FIG. 1

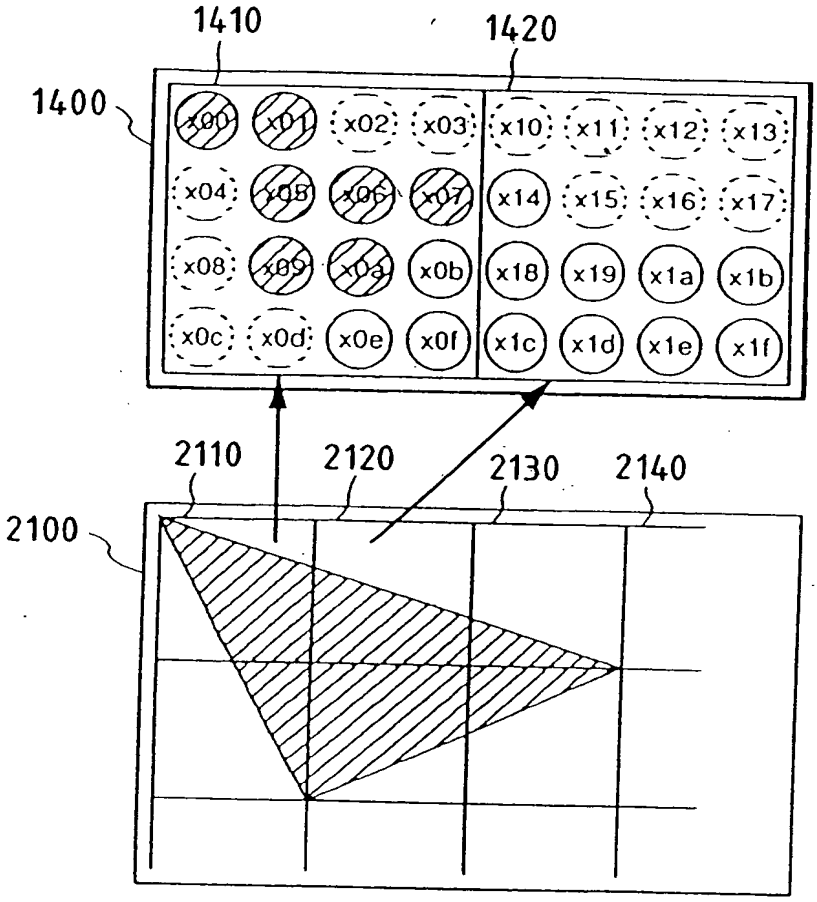

1410
1420
1400
x00
x01
x02
x03
x10
x11
x12
x13
x04
x05
x06
x07
x14
x15
x16
x17
x08
x09
x0a
x0b
x18
x19
x1a
x1b
x0c
x0d
x0e
x0f
x1c
x1d
x1e
x1f
2110
2120
2130
2140
2100

FIG. 2

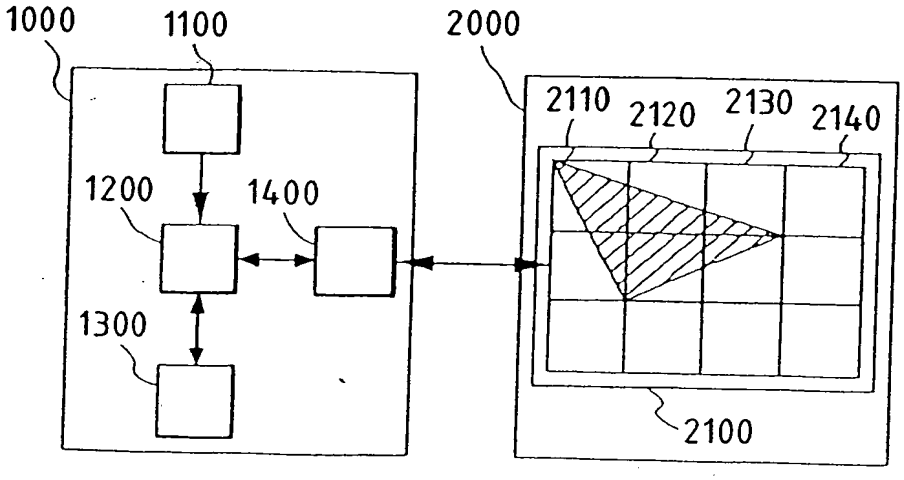

1000
1100
2000
2110
2120
2130
2140
1200
1400
1300
2100

FIG. 3

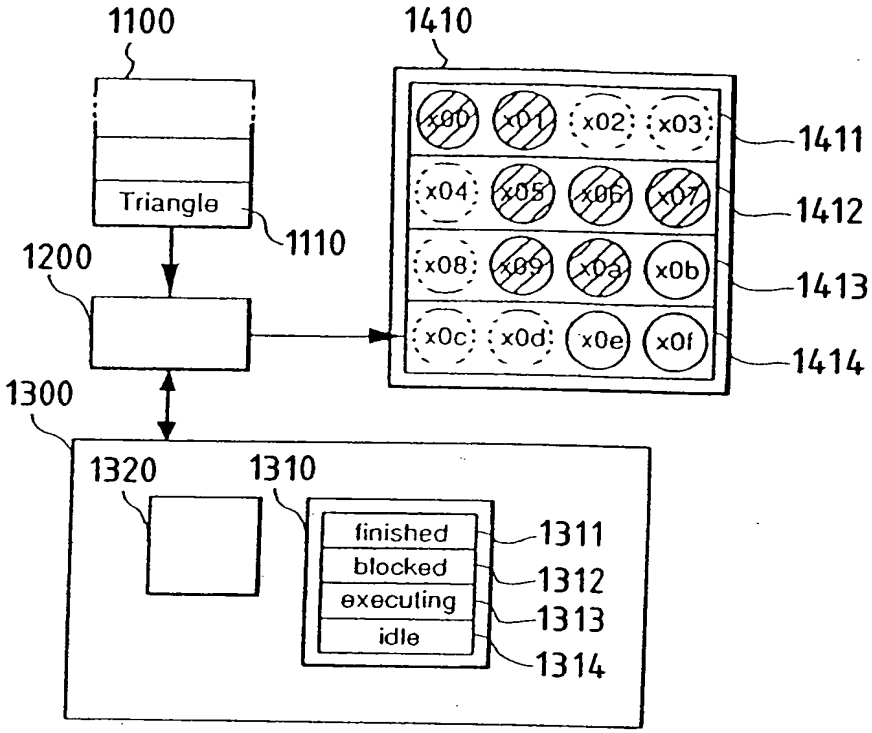


FIG. 4

1320	Xs, Ys, Zs, Rs, Gs, Bs, Y ΔXy, ΔZy, ΔRy, ΔGy, ΔBy ΔZx, ΔRx, ΔGx, ΔBx

1310	X	Y	Z	R	G	B	Flag
1311	1	0	1	1	2	1	111
1312	3	1	4	2	5	5	110
1313	2	2	4	0	2	6	101
1314	?	3	?	?	?	?	000

Flag
000 ··· idle
100 ··· waiting
101 ··· executing
110 ··· blocked
111 ··· finished

FIG. 5

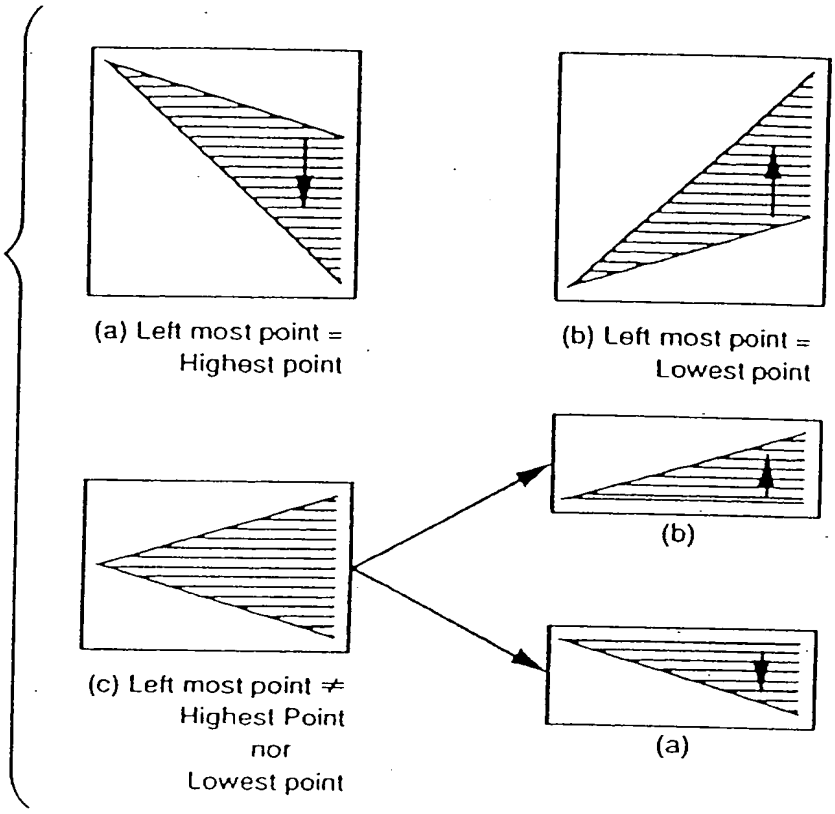
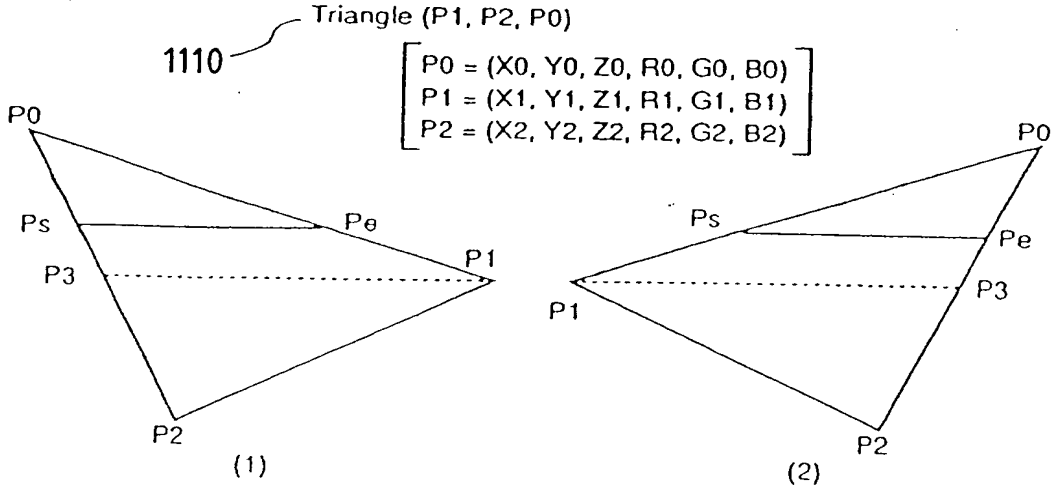


FIG. 6



Triangle (P1, P2, P0) → Triangle (P0, P3, P1) and Triangle (P1, P3, P2)

$P3 = P0 + (Y1 - Y0) / (Y2 - Y0) \cdot (P2 - P0)$

FIG. 7

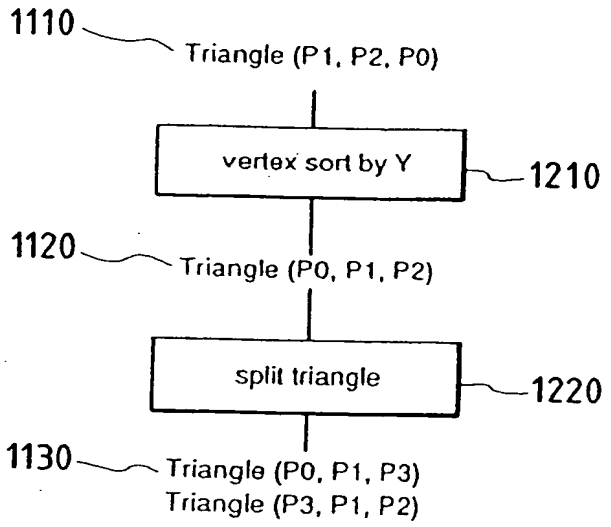

1110
Triangle (P1, P2, P0)
vertex sort by Y
1210
1120
Triangle (P0, P1, P2)
split triangle
1220
1130
Triangle (P0, P1, P3)
Triangle (P3, P1, P2)

FIG. 8

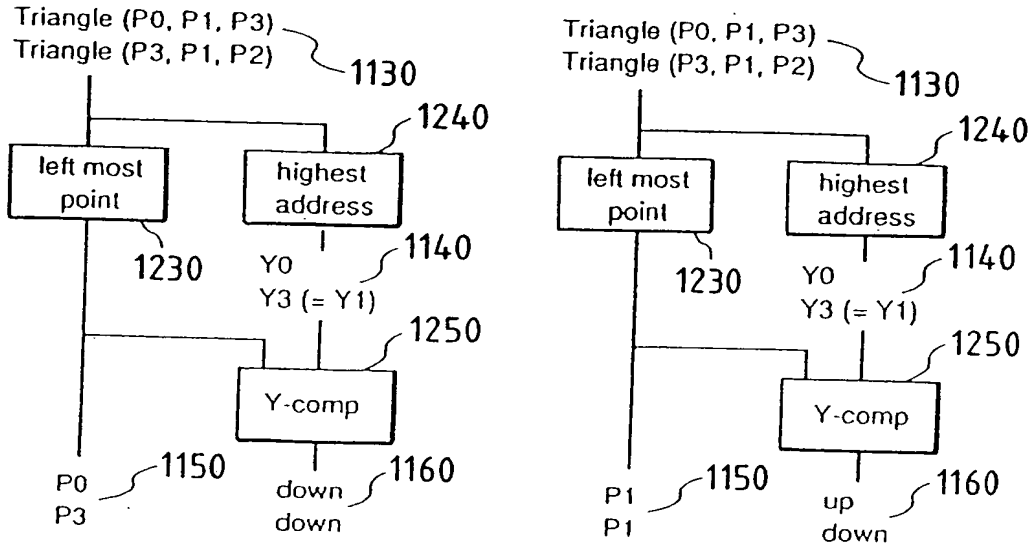

Triangle (P0, P1, P3)
Triangle (P3, P1, P2)
1130
1240
left most point
highest address
1230
Y0
Y3 (= Y1)
1140
1250
Y-comp
P0
P3
1150
down
down
1160
(1)
Triangle (P0, P1, P3)
Triangle (P3, P1, P2)
1130
1240
left most point
highest address
1230
Y0
Y3 (= Y1)
1140
1250
Y-comp
P1
P1
1150
up
down
1160
(2)

FIG. 9

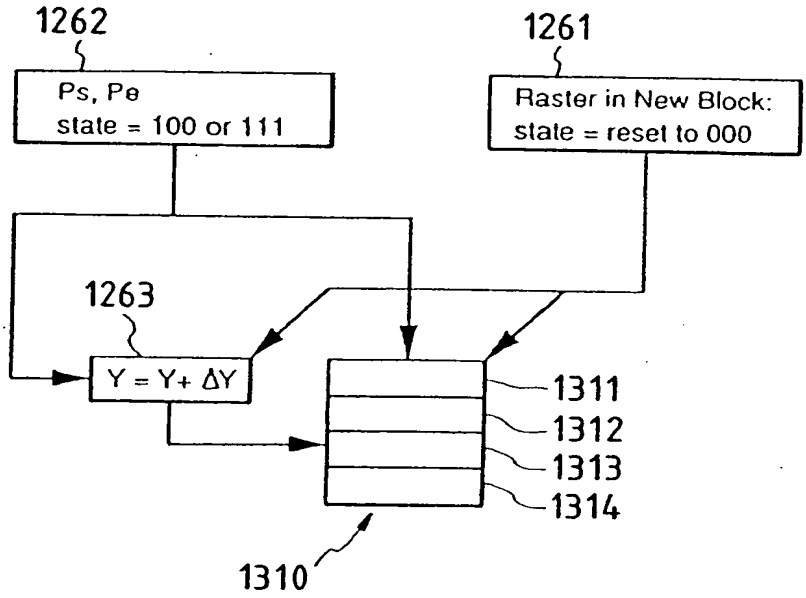

1262
Ps, Pe
state = 100 or 111
1261
Raster in New Block:
state = reset to 000
1263
Y = Y+ ΔY
1311
1312
1313
1314
1310

FIG. 10

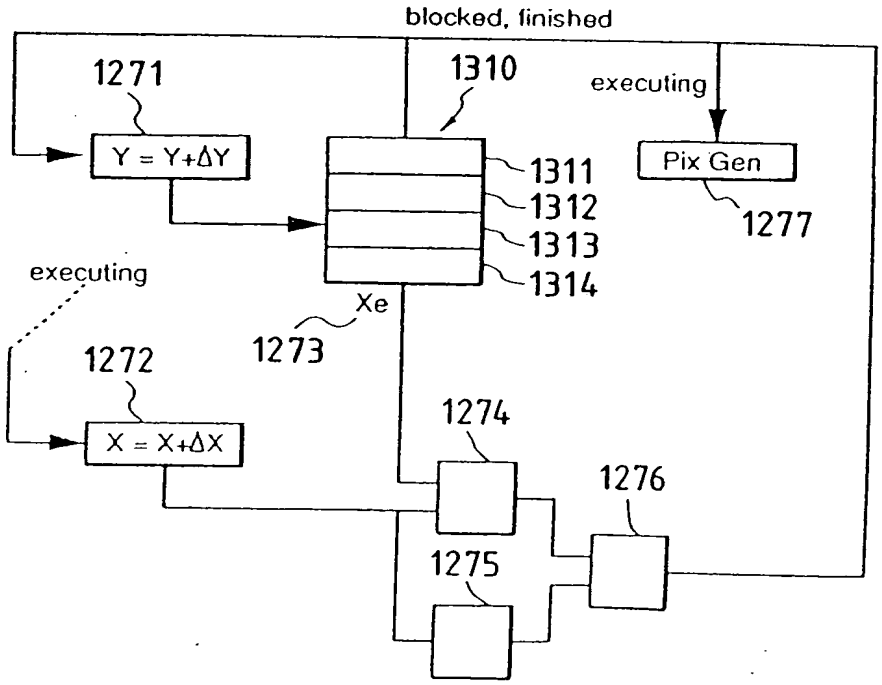

blocked, finished
1271
Y = Y+ΔY
1310
executing
Pix Gen
1277
1311
1312
1313
1314
executing
Xe
1273
1272
X = X+ΔX
1274
1276
1275

FIG. 11

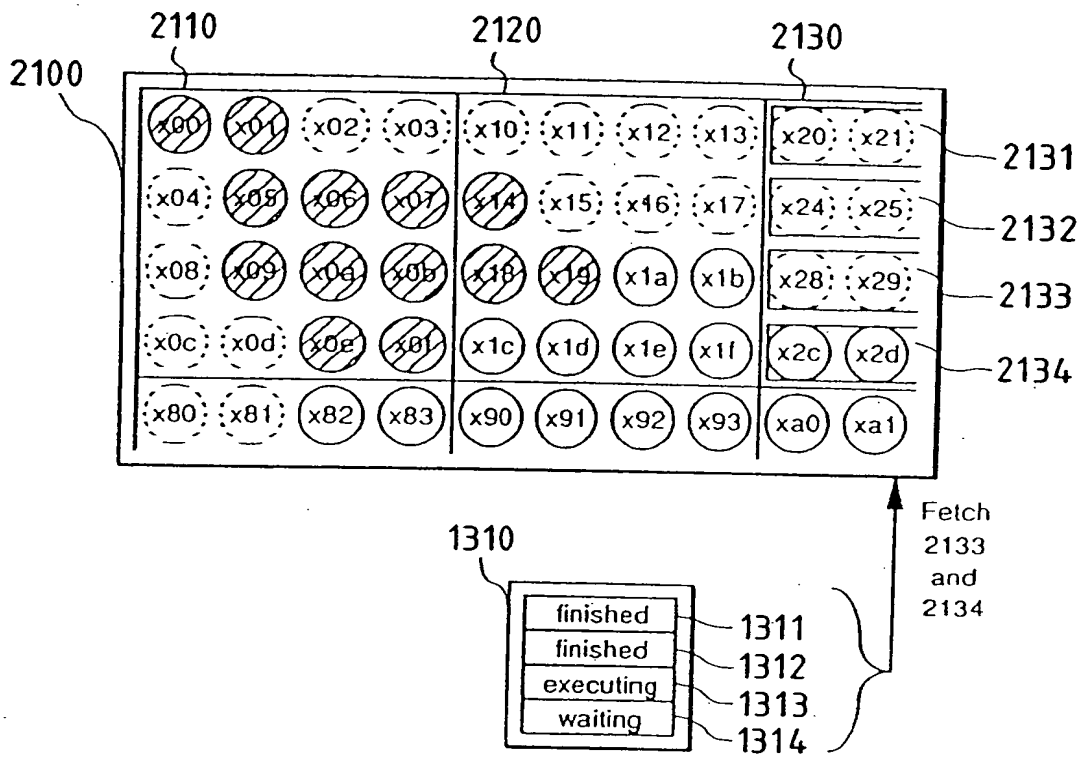

2110
2120
2130
2100
x00
x01
x02
x03
x10
x11
x12
x13
x20
x21
2131
x04
x05
x06
x07
x14
x15
x16
x17
x24
x25
2132
x08
x09
x0a
x0b
x18
x19
x1a
x1b
x28
x29
2133
x0c
x0d
x0e
x0f
x1c
x1d
x1e
x1f
x2c
x2d
2134
x80
x81
x82
x83
x90
x91
x92
x93
xa0
xa1
Fetch
2133
and
2134
1310
finished
finished
executing
waiting
1311
1312
1313
1314

FIG. 12

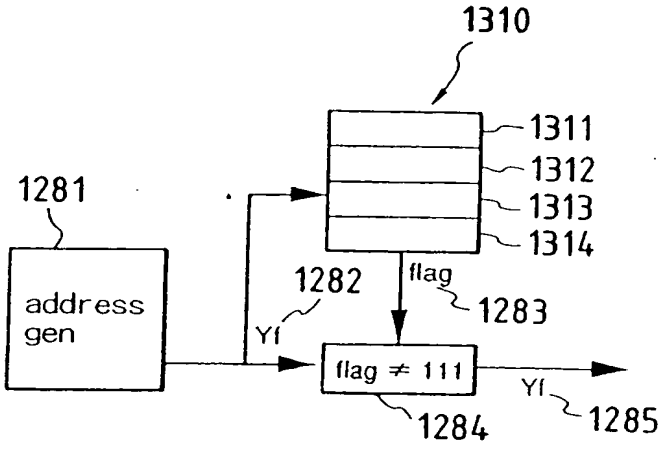

1310
1311
1312
1313
1314
1281
address
gen
1282
Yf
flag
1283
flag ≠ 111
Yf
1284
1285

FIG. 13

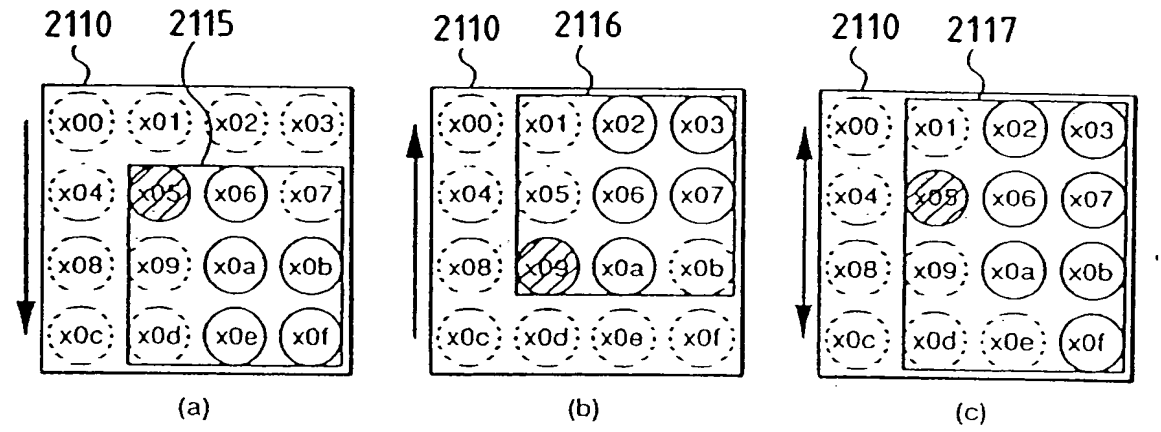

2110
2115
x00
x01
x02
x03
x04
x05
x06
x07
x08
x09
x0a
x0b
x0c
x0d
x0e
x0f
(a)
2110
2116
(b)
2110
2117
(c)

FIG. 14

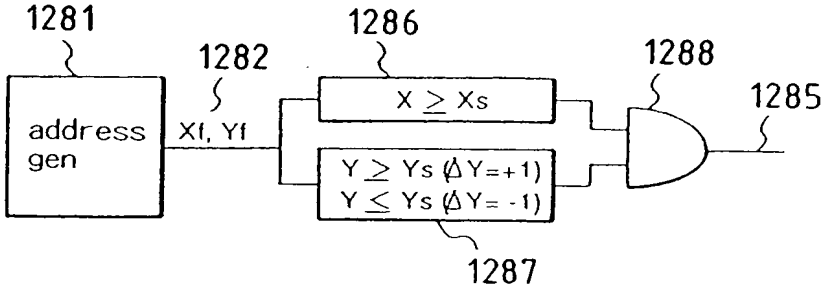

1281
address gen
1282
Xf, Yf
1286
X ≥ Xs
Y ≥ Ys (ΔY=+1)
Y ≤ Ys (ΔY=-1)
1287
1288
1285

FIG. 15

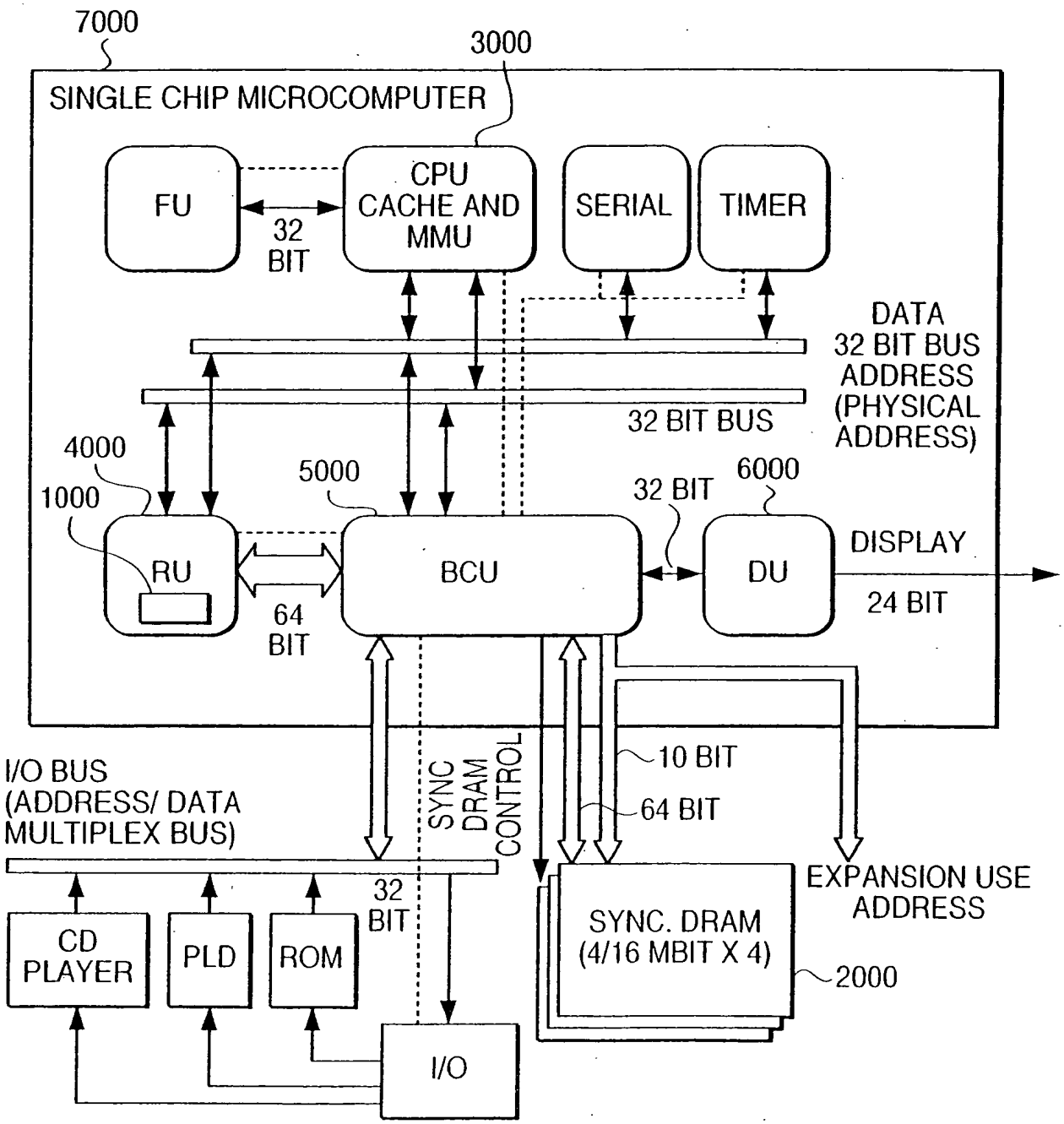

7000
3000
SINGLE CHIP MICROCOMPUTER
FU
32 BIT
CPU CACHE AND MMU
SERIAL
TIMER
DATA 32 BIT BUS ADDRESS (PHYSICAL ADDRESS)
32 BIT BUS
4000
1000
5000
32 BIT
6000
RU
BCU
DU
DISPLAY
24 BIT
64 BIT
I/O BUS (ADDRESS/ DATA MULTIPLEX BUS)
SYNC DRAM CONTROL
10 BIT
64 BIT
EXPANSION USE ADDRESS
32 BIT
CD PLAYER
PLD
ROM
SYNC. DRAM (4/16 MBIT X 4)
2000
I/O

FIG. 16

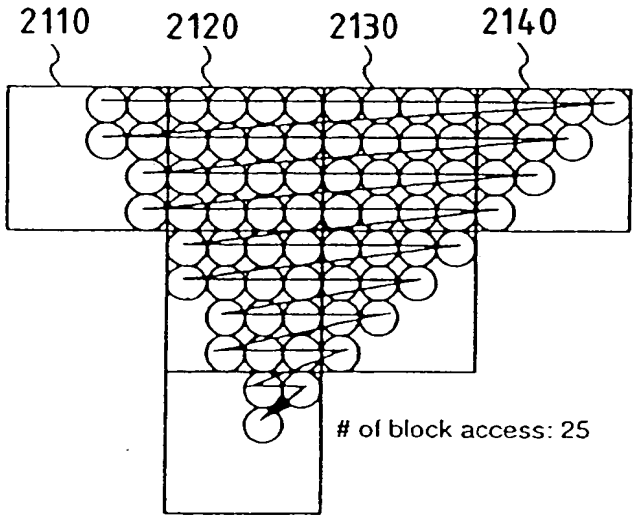


FIG. 17
(PRIOR ART)

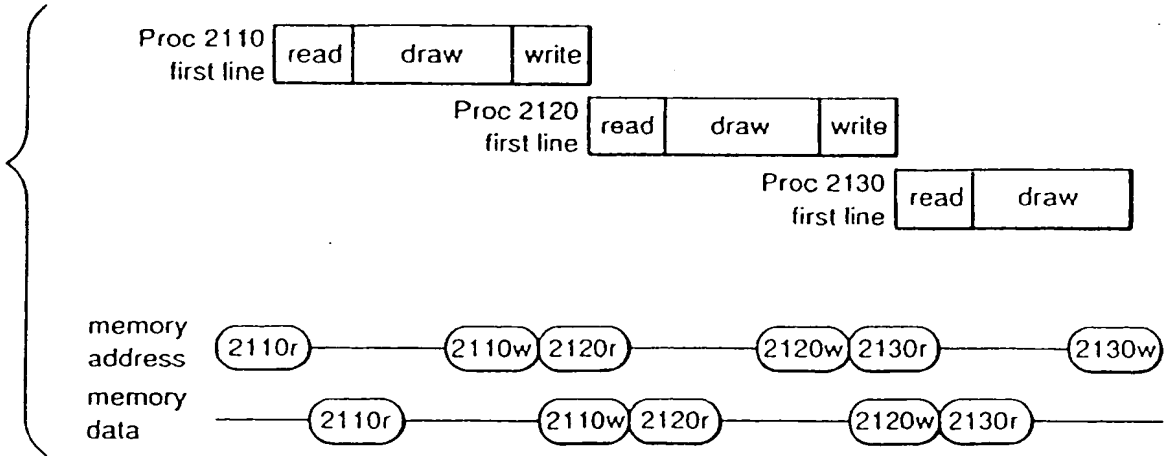


FIG. 18

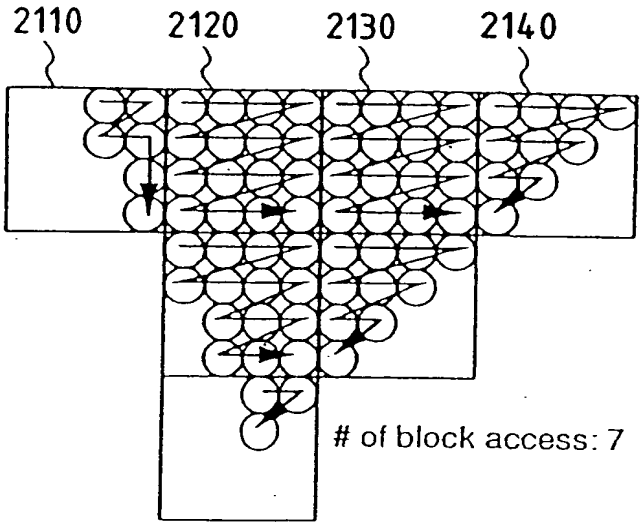

2110
2120
2130
2140
# of block access: 7

FIG. 19

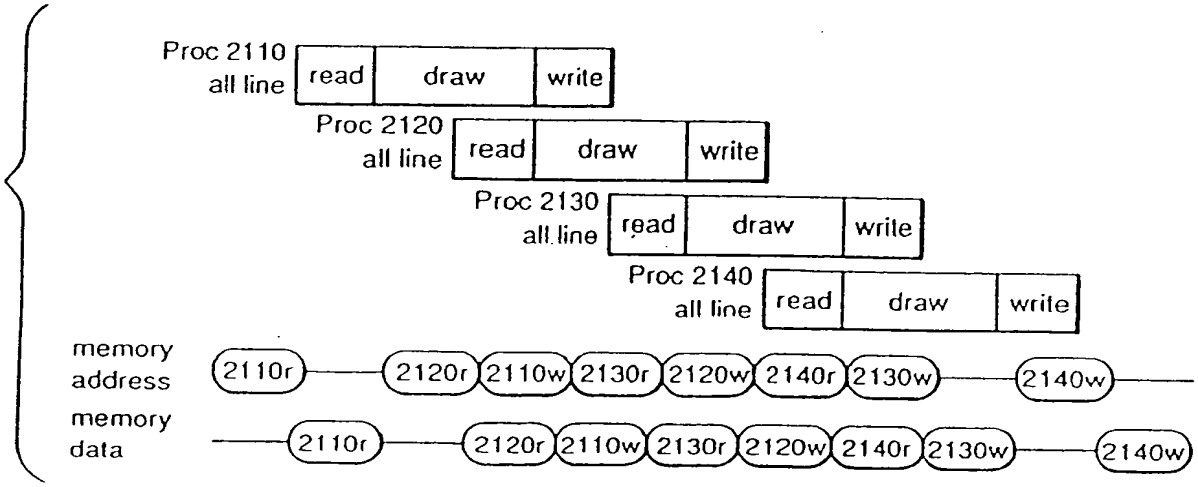

Proc 2110 all line
read
draw
write
Proc 2120 all line
read
draw
write
Proc 2130 all line
read
draw
write
Proc 2140 all line
read
draw
write
memory address
2110r
2120r
2110w
2130r
2120w
2140r
2130w
2140w
memory data
2110r
2120r
2110w
2130r
2120w
2140r
2130w
2140w

FIG. 20

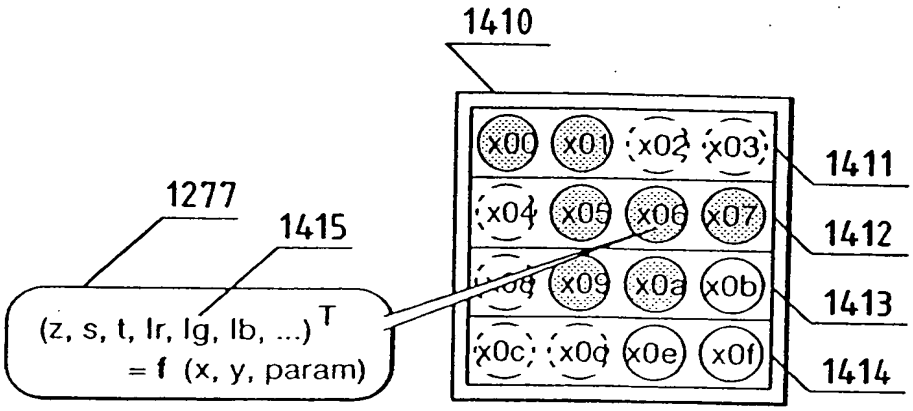


FIG. 21

1320	param, current block, pivot block, direction, next pivot

1310	Xs	Xe	Lflag	Rflag
1311	x00	x01	111	111
1312	x05	x14	111	110
1313	x09	x18	111	101
1314	?	?	000	000

Flag
000 ··· idle
100 ··· waiting
101 ··· executing
110 ··· blocked
111 ··· finished

FIG. 22

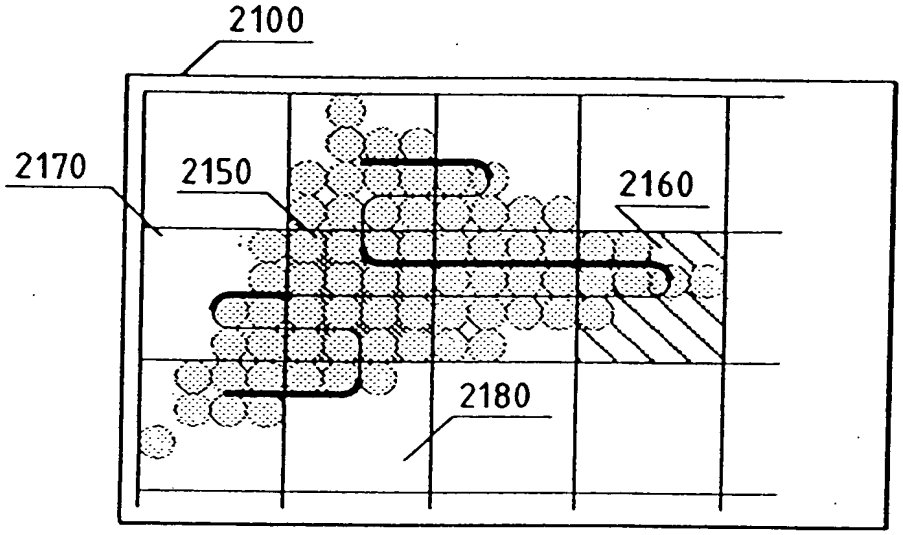

2100
2170
2150
2160
2180

FIG. 23

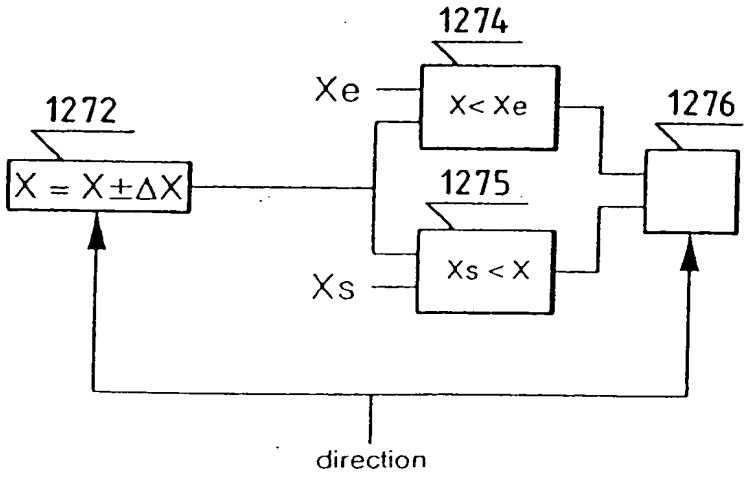

1274
Xe
X< Xe
1272
1276
X = X±ΔX
1275
Xs < X
Xs
direction

FIG. 24

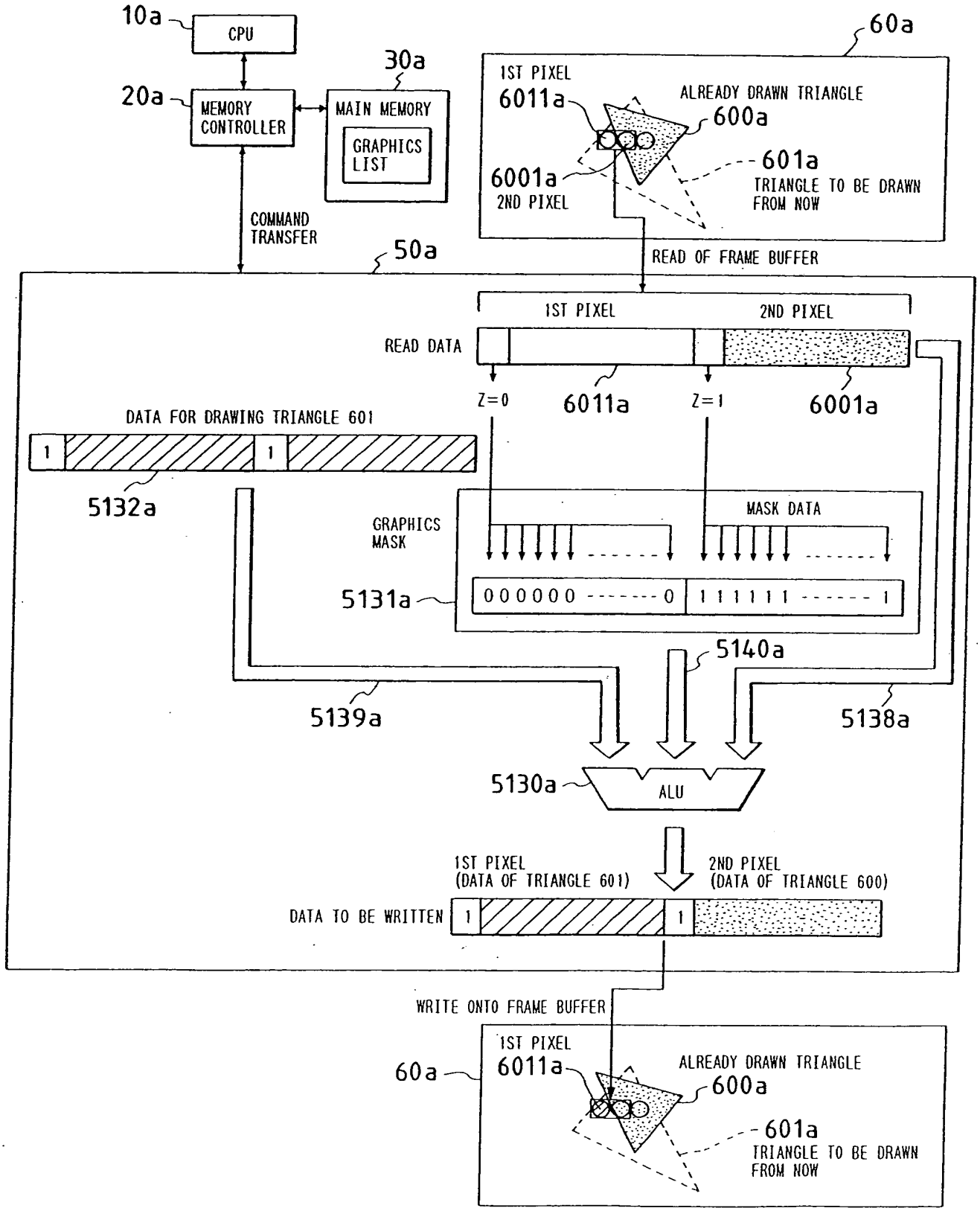

10a
CPU
20a
MEMORY CONTROLLER
30a
MAIN MEMORY
GRAPHICS LIST
COMMAND TRANSFER
60a
1ST PIXEL
6011a
ALREADY DRAWN TRIANGLE
600a
601a
TRIANGLE TO BE DRAWN FROM NOW
6001a
2ND PIXEL
50a
READ OF FRAME BUFFER
1ST PIXEL
2ND PIXEL
READ DATA
Z=0
6011a
Z=1
6001a
DATA FOR DRAWING TRIANGLE 601
1
1
5132a
GRAPHICS MASK
MASK DATA
5131a
000000 ------- 0
111111 ------ 1
5140a
5139a
5138a
5130a
ALU
1ST PIXEL (DATA OF TRIANGLE 601)
2ND PIXEL (DATA OF TRIANGLE 600)
DATA TO BE WRITTEN
1
1
WRITE ONTO FRAME BUFFER
60a
1ST PIXEL
6011a
ALREADY DRAWN TRIANGLE
600a
601a
TRIANGLE TO BE DRAWN FROM NOW

FIG. 25

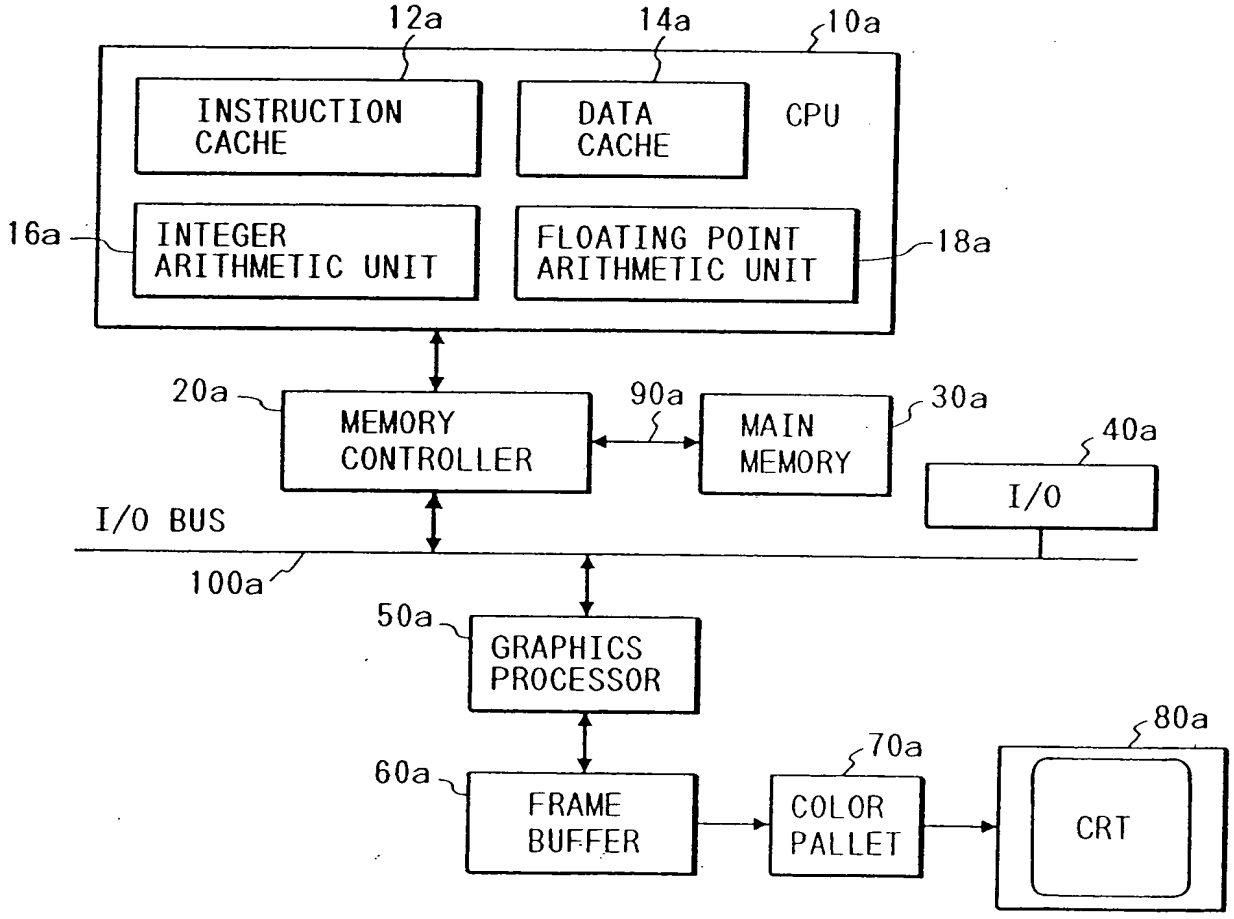

12a
14a
10a
INSTRUCTION CACHE
DATA CACHE
CPU
16a
INTEGER ARITHMETIC UNIT
FLOATING POINT ARITHMETIC UNIT
18a
20a
MEMORY CONTROLLER
90a
MAIN MEMORY
30a
40a
I/O
I/O BUS
100a
50a
GRAPHICS PROCESSOR
60a
FRAME BUFFER
70a
COLOR PALLET
80a
CRT

FIG. 26

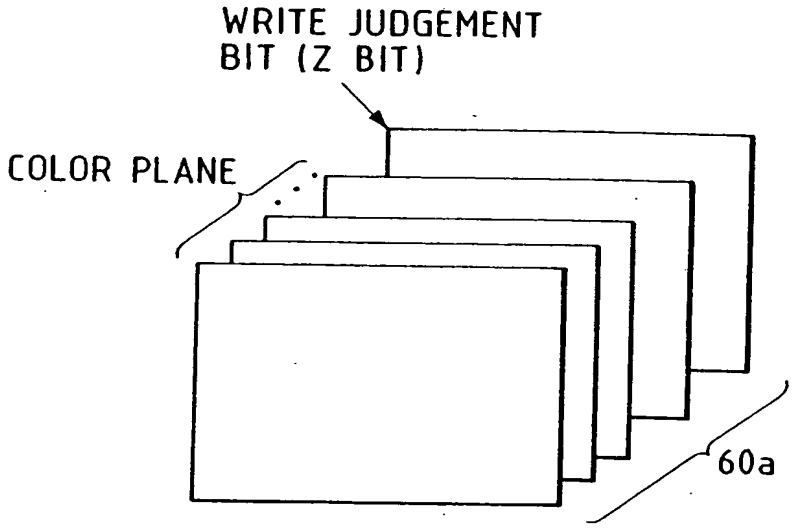

WRITE JUDGEMENT
BIT (Z BIT)
COLOR PLANE
60a

FIG. 27

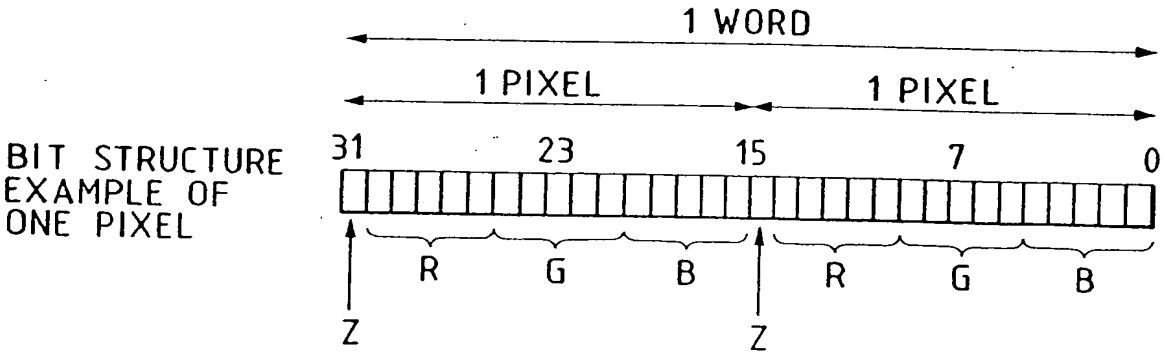

1 WORD
1 PIXEL
1 PIXEL
BIT STRUCTURE
EXAMPLE OF
ONE PIXEL
31
23
15
7
0
R
G
B
R
G
B
Z
Z

FIG. 28

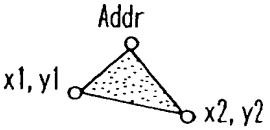
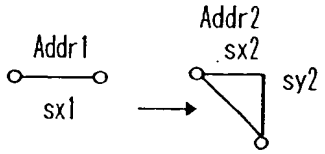
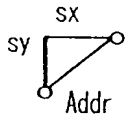
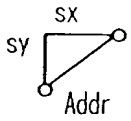
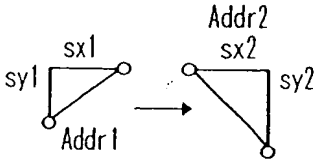
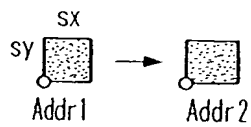
COMMAND	OPERATION CONTENTS
3D TRIANGLE DRAW USED FOR DRAWING POLYGON WITH GOURAUD SHADING	TRIAGL Addr, x1, y1 x2, y2, p, dpx, dpy  Addr:MEMORY ADDRESS OF DRAW STARTING POINT x1, y1:RELATIVE COORDINATE APEX FROM Addr x2, y2:RELATIVE COORDINATE APEX FROM Addr p:DRAW DATA OF DRAW STARTING POINT dpx:VARIED COMPONENT OF 1 DOT IN X DIRECTION dpy:VARIED COMPONENT OF 1 DOT IN Y DIRECTION
3D STRAIGHT LINE COPY USED FOR DRAWING POLYGON WITH TEXTURE MAPPING	CPLINE3 Addr1, sx1, Addr2, sx2, Sy2  Addr1:MEMORY ADDRESS OF REFERENCE START POINT sx1:X DIRECTION DOT NUMBER OF REFERENCE LINE Addr2:MEMORY ADDRESS OF DRAW STARTING POINT sx2:X DIRECTION DOT NUMBER OF DRAW LINE sy2:Y DIRECTION DOT NUMBER OF DRAW LINE
3D STRAIGHT LINE DRAW	LINE3 Addr, sx, sy, p, dp  Addr:MEMORY ADDRESS OF DRAW STARTING POINT sx:X DIRECTION DOT NUMBER OF DRAW LINE sy:Y DIRECTION DOT NUMBER OF DRAW LINE p:DRAW DATA OF DRAW STARTING POINT dp:DRAW DATA VARIED COMPONENT OF 1 DOT
2D STRAIGHT LINE DRAW	LINE2 Addr, sx, sy, p  Addr:MEMORY ADDRESS OF DRAW STARTING POINT sx:X DIRECTION DOT NUMBER OF DRAW LINE sy:Y DIRECTION DOT NUMBER OF DRAW LINE p:DRAW DATA OF DRAW STARTING POINT
2D STRAIGHT LINE COPY WITH ENLARGEMENT, REDUCTION AND ROTATION	CPLINE3 Addr1, sx1, sy1, Addr2, sx2, Sy2  Addr1:MEMORY ADDRESS OF REFERENCE START POINT sx1:X DIRECTION DOT NUMBER OF REFERENCE LINE sy1:Y DIRECTION DOT NUMBER OF REFERENCE LINE Addr2:MEMORY ADDRESS OF DRAW STARTING POINT sx2:X DIRECTION DOT NUMBER OF DRAW LINE sy2:Y DIRECTION DOT NUMBER OF DRAW LINE
BitBLT	BITBLT Addr1, sx, sy, Addr2  Addr1:MEMORY ADDRESS OF REFERENCE START POINT sx:X DIRECTION DOT NUMBER OF REFERENCE LINE sy:Y DIRECTION DOT NUMBER OF REFERENCE LINE Addr2:MEMORY ADDRESS OF DRAW STARTING POINT

FIG. 29

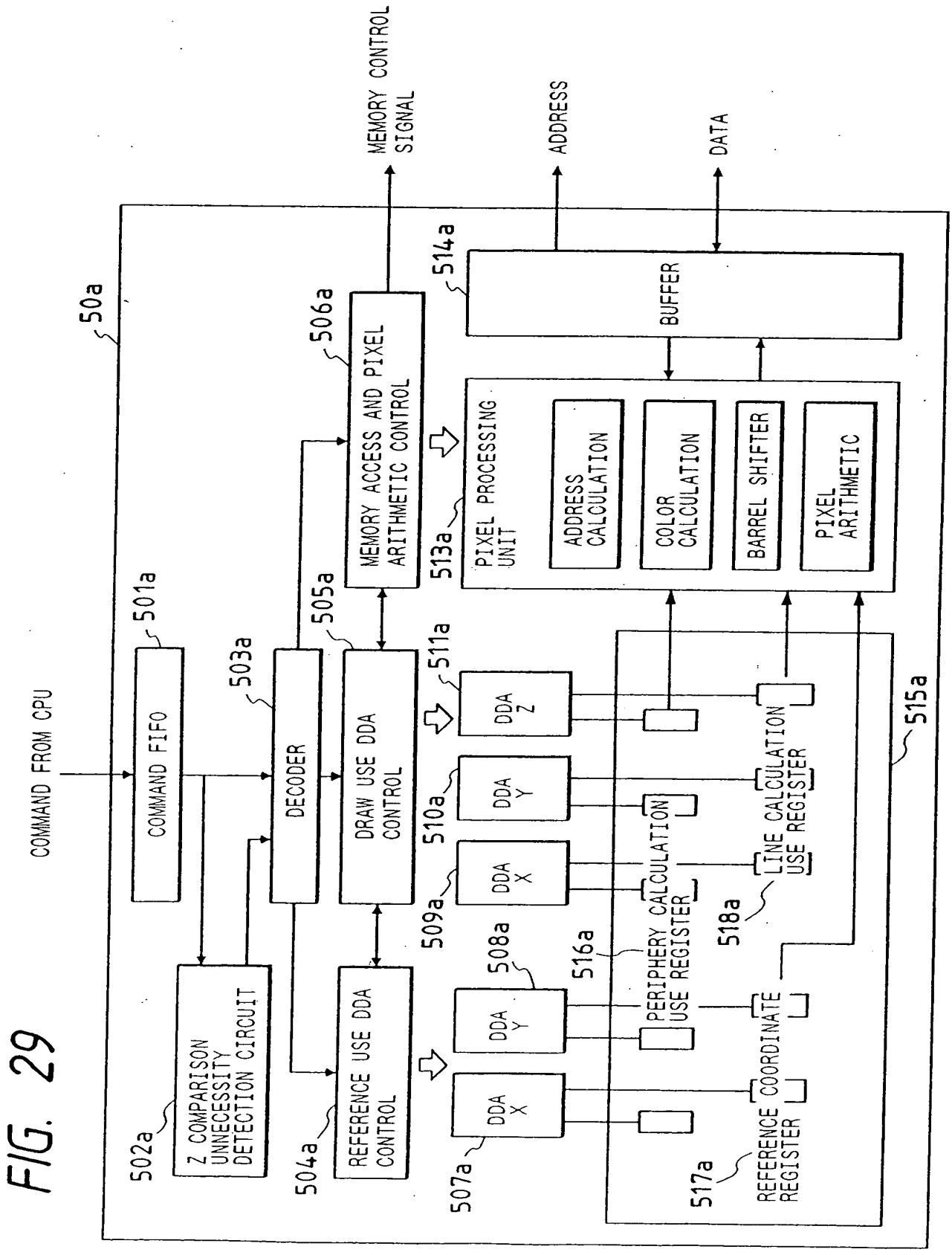

COMMAND FROM CPU
50a
501a
COMMAND FIFO
503a
DECODER
505a
DRAW USE DDA CONTROL
506a
MEMORY ACCESS AND PIXEL ARITHMETIC CONTROL
MEMORY CONTROL SIGNAL
ADDRESS
DATA
514a
BUFFER
513a
PIXEL PROCESSING UNIT
ADDRESS CALCULATION
COLOR CALCULATION
BARREL SHIFTER
PIXEL ARITHMETIC
511a
DDA Z
510a
DDA Y
509a
DDA X
508a
DDA Y
507a
DDA X
502a
Z COMPARISON UNNECESSITY DETECTION CIRCUIT
504a
REFERENCE USE DDA CONTROL
516a
PERIPHERY CALCULATION USE REGISTER
518a
LINE CALCULATION USE REGISTER
515a
517a
REFERENCE COORDINATE REGISTER

FIG. 30

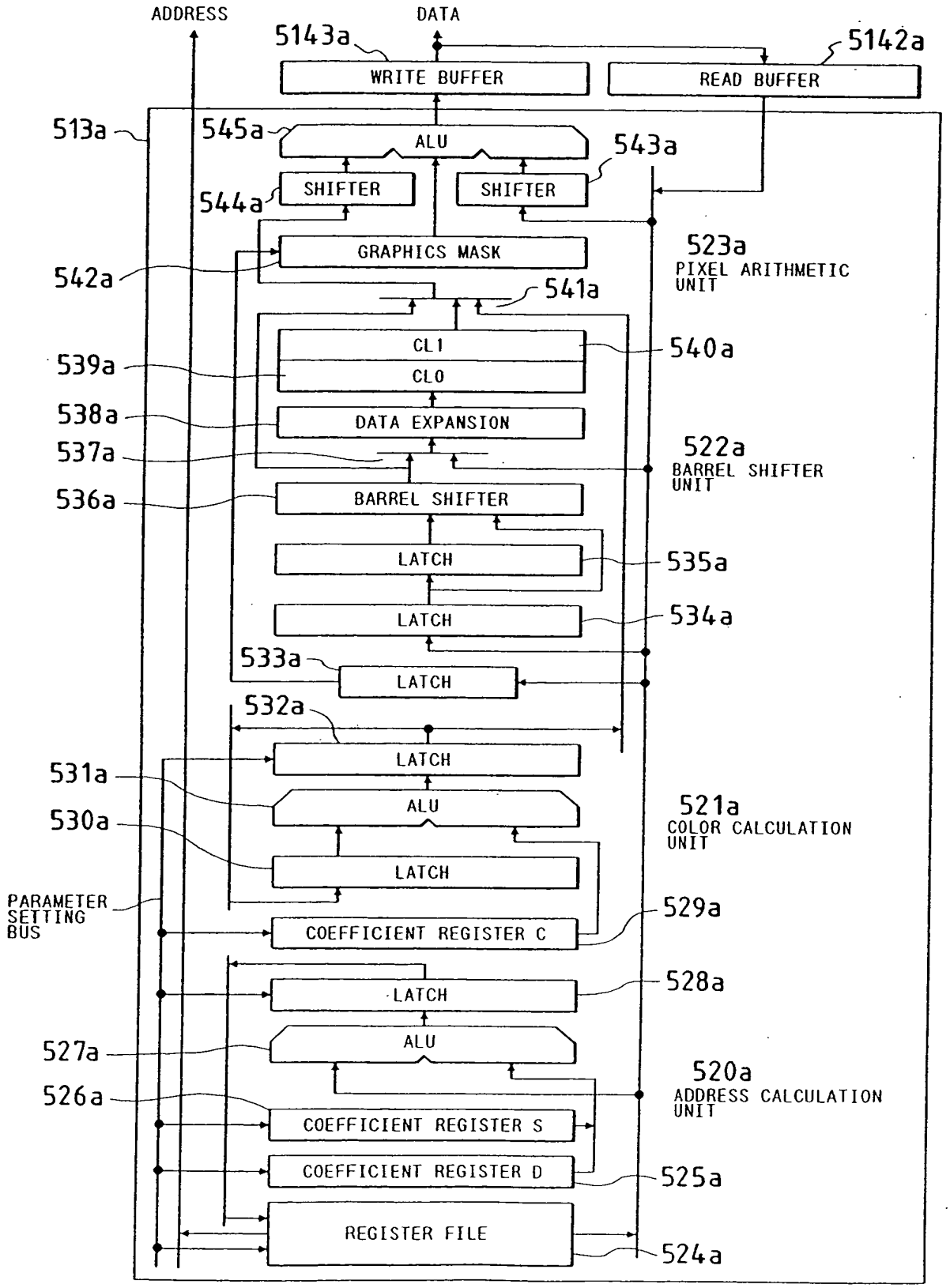

ADDRESS
DATA
5143a
5142a
WRITE BUFFER
READ BUFFER
513a
545a
ALU
543a
SHIFTER
SHIFTER
544a
GRAPHICS MASK
523a
PIXEL ARITHMETIC UNIT
542a
541a
CL1
540a
539a
CL0
538a
DATA EXPANSION
537a
522a
BARREL SHIFTER UNIT
536a
BARREL SHIFTER
LATCH
535a
LATCH
534a
533a
LATCH
532a
LATCH
531a
ALU
521a
COLOR CALCULATION UNIT
530a
LATCH
PARAMETER SETTING BUS
COEFFICIENT REGISTER C
529a
LATCH
528a
527a
ALU
520a
ADDRESS CALCULATION UNIT
526a
COEFFICIENT REGISTER S
COEFFICIENT REGISTER D
525a
REGISTER FILE
524a

FIG. 31

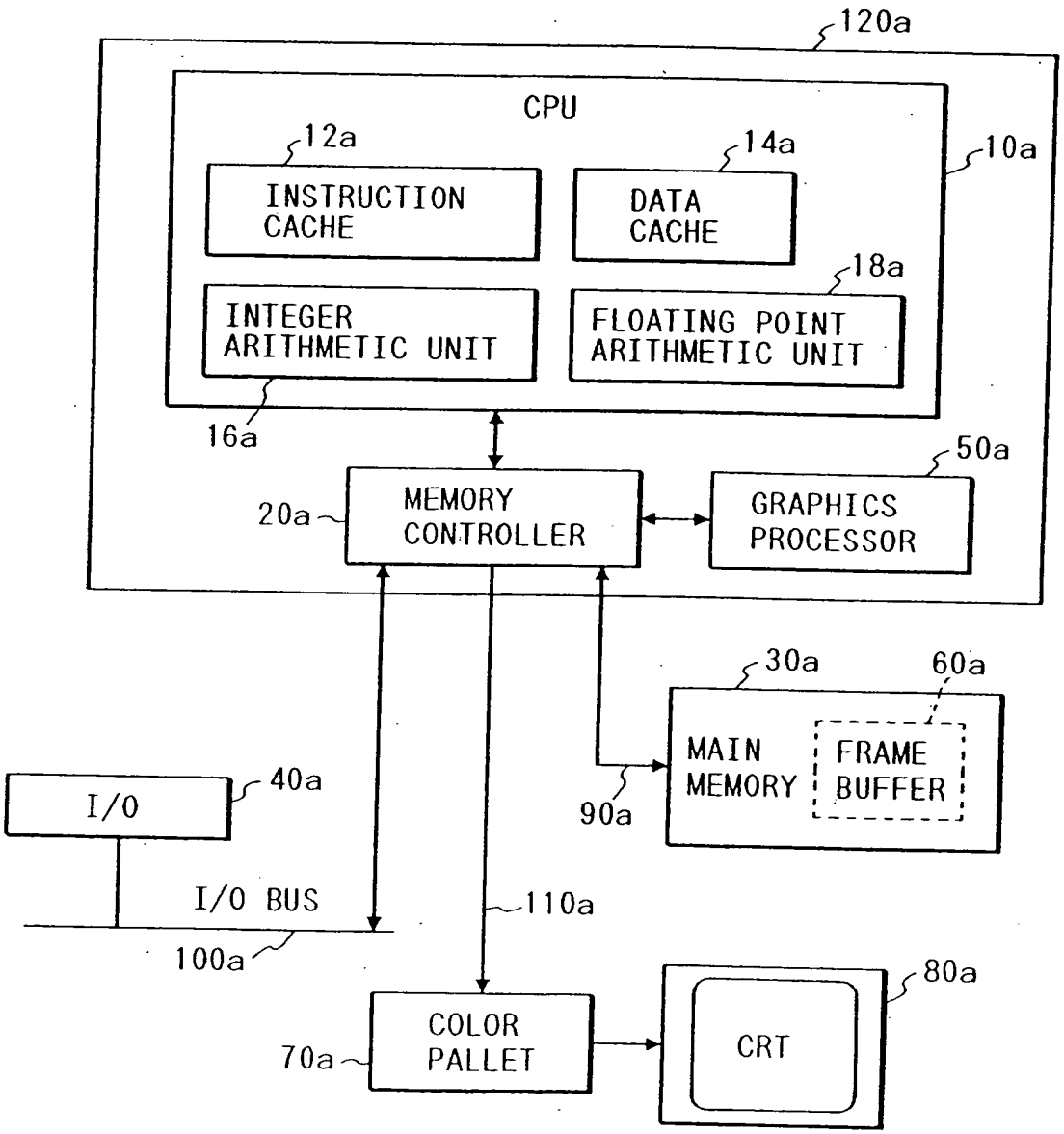

120a
CPU
12a
INSTRUCTION CACHE
14a
DATA CACHE
10a
18a
INTEGER ARITHMETIC UNIT
FLOATING POINT ARITHMETIC UNIT
16a
50a
20a
MEMORY CONTROLLER
GRAPHICS PROCESSOR
30a
60a
MAIN MEMORY
FRAME BUFFER
90a
I/O
40a
I/O BUS
110a
100a
80a
COLOR PALLET
70a
CRT

FIG. 32

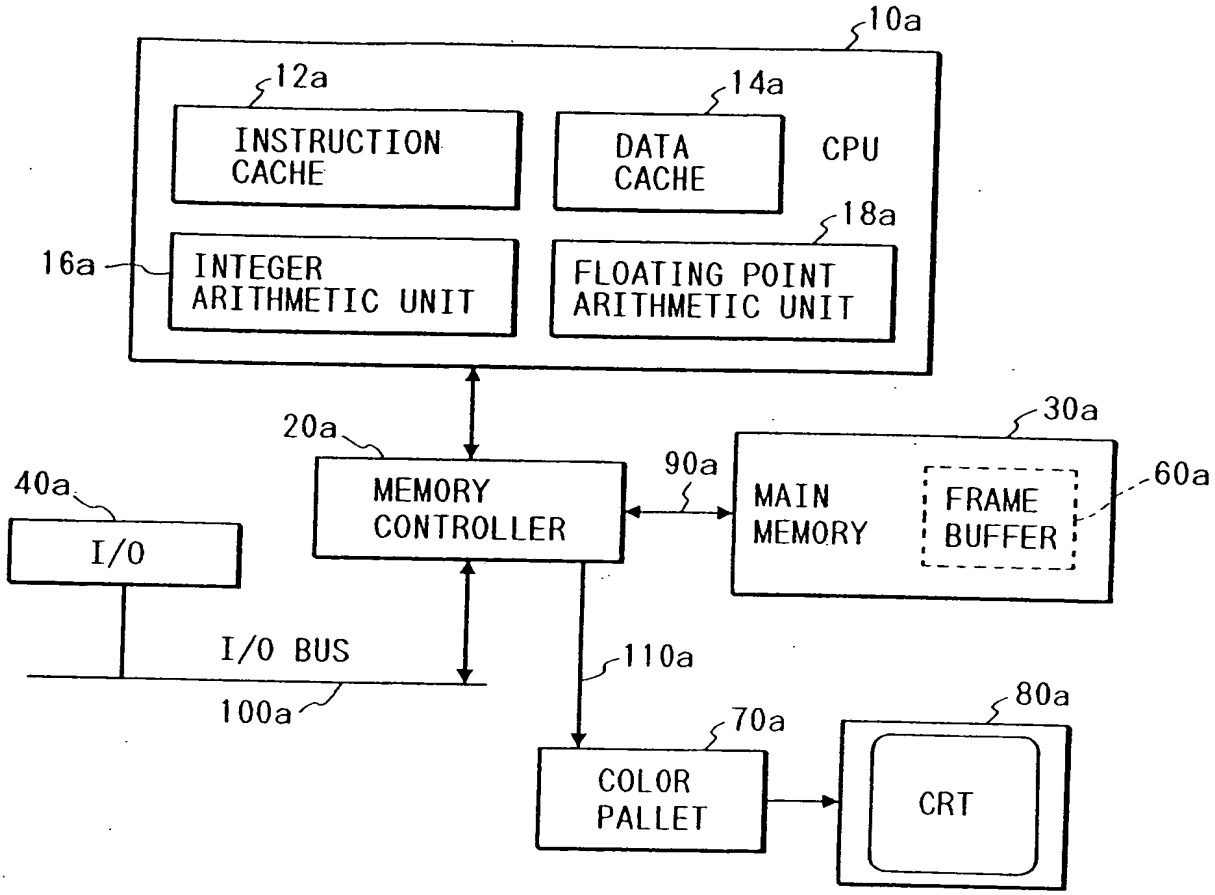

10a
12a
INSTRUCTION CACHE
14a
DATA CACHE
CPU
18a
16a
INTEGER ARITHMETIC UNIT
FLOATING POINT ARITHMETIC UNIT
20a
30a
90a
MEMORY CONTROLLER
MAIN MEMORY
FRAME BUFFER
60a
40a
I/O
I/O BUS
110a
100a
80a
70a
COLOR PALLET
CRT

FIG. 33

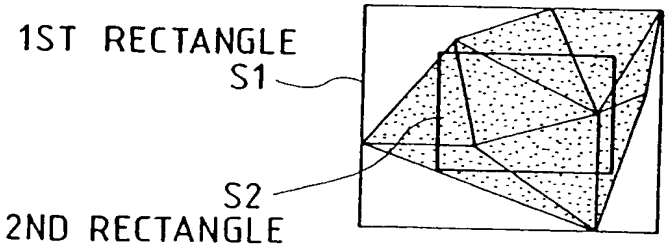

1ST RECTANGLE
S1
S2
2ND RECTANGLE

FIG. 34

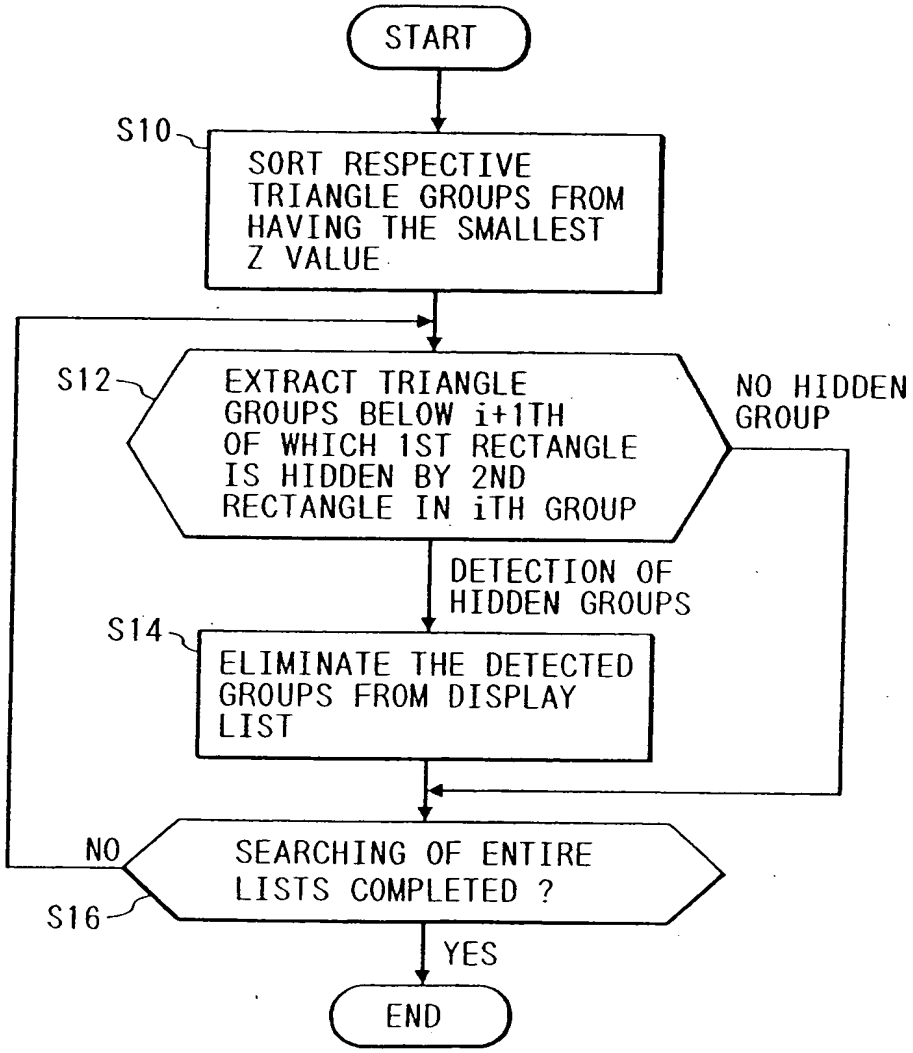

START
S10
SORT RESPECTIVE TRIANGLE GROUPS FROM HAVING THE SMALLEST Z VALUE
S12
EXTRACT TRIANGLE GROUPS BELOW i+1TH OF WHICH 1ST RECTANGLE IS HIDDEN BY 2ND RECTANGLE IN iTH GROUP
NO HIDDEN GROUP
DETECTION OF HIDDEN GROUPS
S14
ELIMINATE THE DETECTED GROUPS FROM DISPLAY LIST
NO
SEARCHING OF ENTIRE LISTS COMPLETED ?
S16
YES
END

FIG. 35

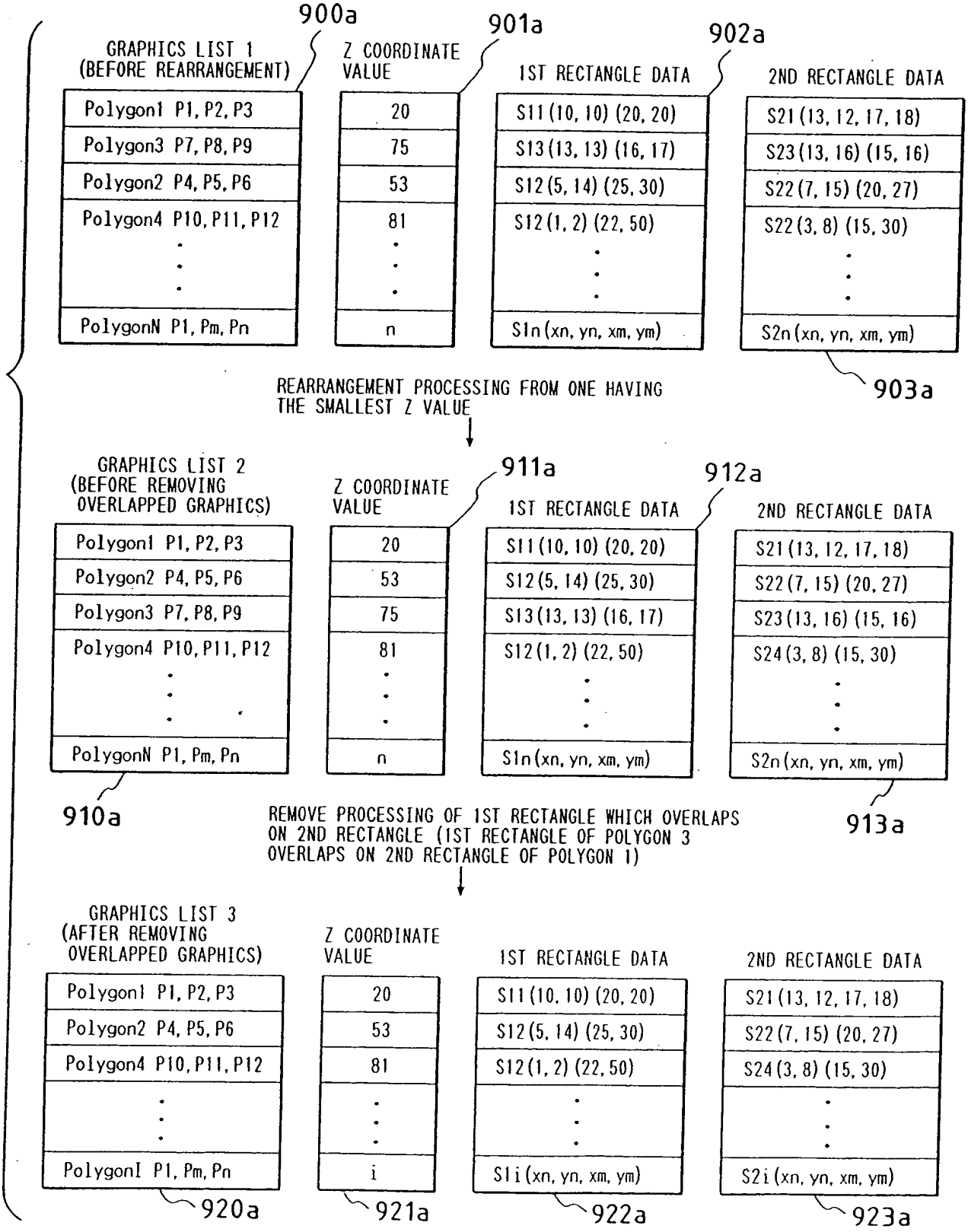

900a
GRAPHICS LIST 1 (BEFORE REARRANGEMENT)
Polygon1 P1, P2, P3
Polygon3 P7, P8, P9
Polygon2 P4, P5, P6
Polygon4 P10, P11, P12
PolygonN P1, Pm, Pn
901a
Z COORDINATE VALUE
20
75
53
81
n
902a
1ST RECTANGLE DATA
S11 (10, 10) (20, 20)
S13 (13, 13) (16, 17)
S12 (5, 14) (25, 30)
S12 (1, 2) (22, 50)
S1n (xn, yn, xm, ym)
2ND RECTANGLE DATA
S21 (13, 12, 17, 18)
S23 (13, 16) (15, 16)
S22 (7, 15) (20, 27)
S22 (3, 8) (15, 30)
S2n (xn, yn, xm, ym)
903a
REARRANGEMENT PROCESSING FROM ONE HAVING THE SMALLEST Z VALUE
GRAPHICS LIST 2 (BEFORE REMOVING OVERLAPPED GRAPHICS)
Polygon1 P1, P2, P3
Polygon2 P4, P5, P6
Polygon3 P7, P8, P9
Polygon4 P10, P11, P12
PolygonN P1, Pm, Pn
910a
911a
Z COORDINATE VALUE
20
53
75
81
n
912a
1ST RECTANGLE DATA
S11 (10, 10) (20, 20)
S12 (5, 14) (25, 30)
S13 (13, 13) (16, 17)
S12 (1, 2) (22, 50)
S1n (xn, yn, xm, ym)
2ND RECTANGLE DATA
S21 (13, 12, 17, 18)
S22 (7, 15) (20, 27)
S23 (13, 16) (15, 16)
S24 (3, 8) (15, 30)
S2n (xn, yn, xm, ym)
913a
REMOVE PROCESSING OF 1ST RECTANGLE WHICH OVERLAPS ON 2ND RECTANGLE (1ST RECTANGLE OF POLYGON 3 OVERLAPS ON 2ND RECTANGLE OF POLYGON 1)
GRAPHICS LIST 3 (AFTER REMOVING OVERLAPPED GRAPHICS)
Polygon1 P1, P2, P3
Polygon2 P4, P5, P6
Polygon4 P10, P11, P12
PolygonI P1, Pm, Pn
920a
Z COORDINATE VALUE
20
53
81
i
921a
1ST RECTANGLE DATA
S11 (10, 10) (20, 20)
S12 (5, 14) (25, 30)
S12 (1, 2) (22, 50)
S1i (xn, yn, xm, ym)
922a
2ND RECTANGLE DATA
S21 (13, 12, 17, 18)
S22 (7, 15) (20, 27)
S24 (3, 8) (15, 30)
S2i (xn, yn, xm, ym)
923a

FIG. 36

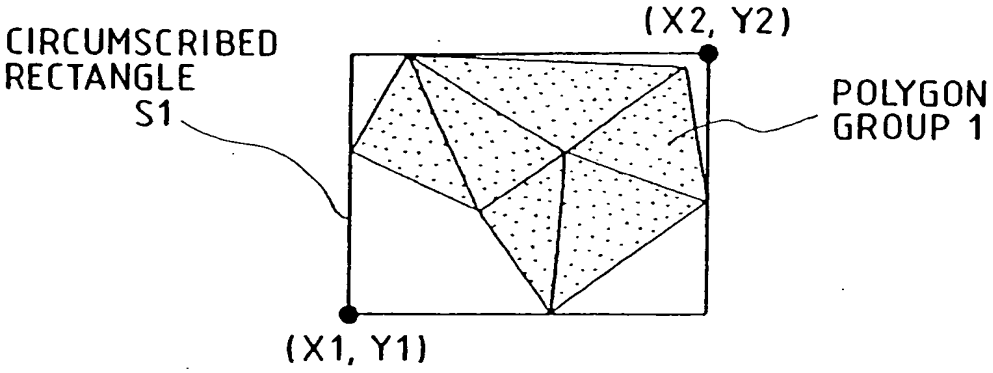

CIRCUMSCRIBED
RECTANGLE
S1
(X2, Y2)
POLYGON
GROUP 1
(X1, Y1)

FIG. 37

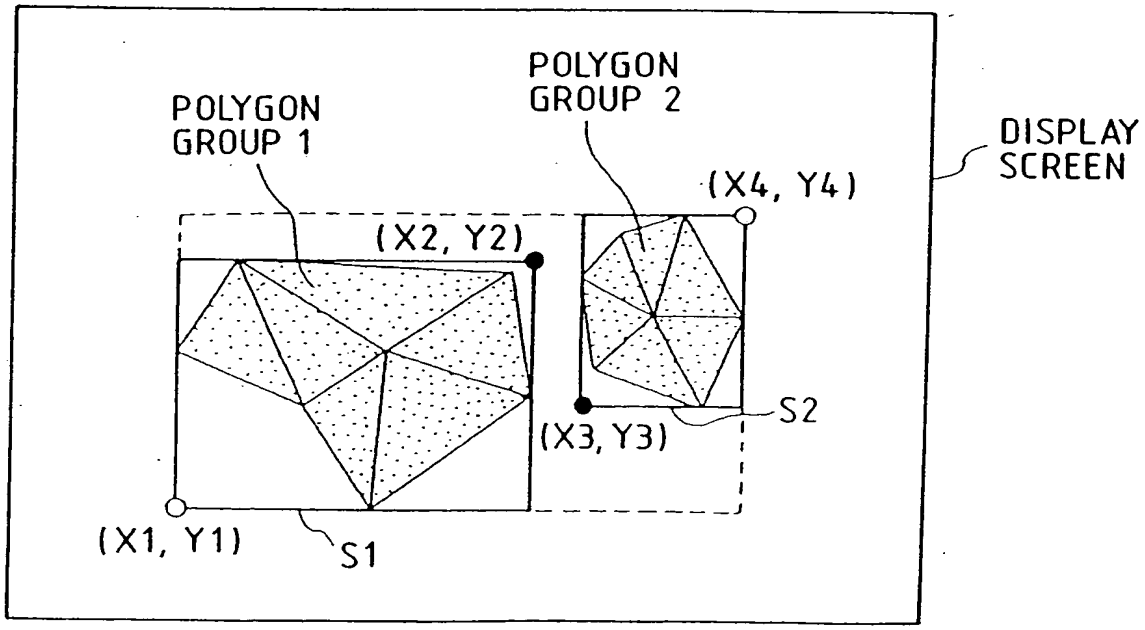

POLYGON
GROUP 1
POLYGON
GROUP 2
DISPLAY
SCREEN
(X4, Y4)
(X2, Y2)
S2
(X3, Y3)
(X1, Y1)
S1

FIG. 38

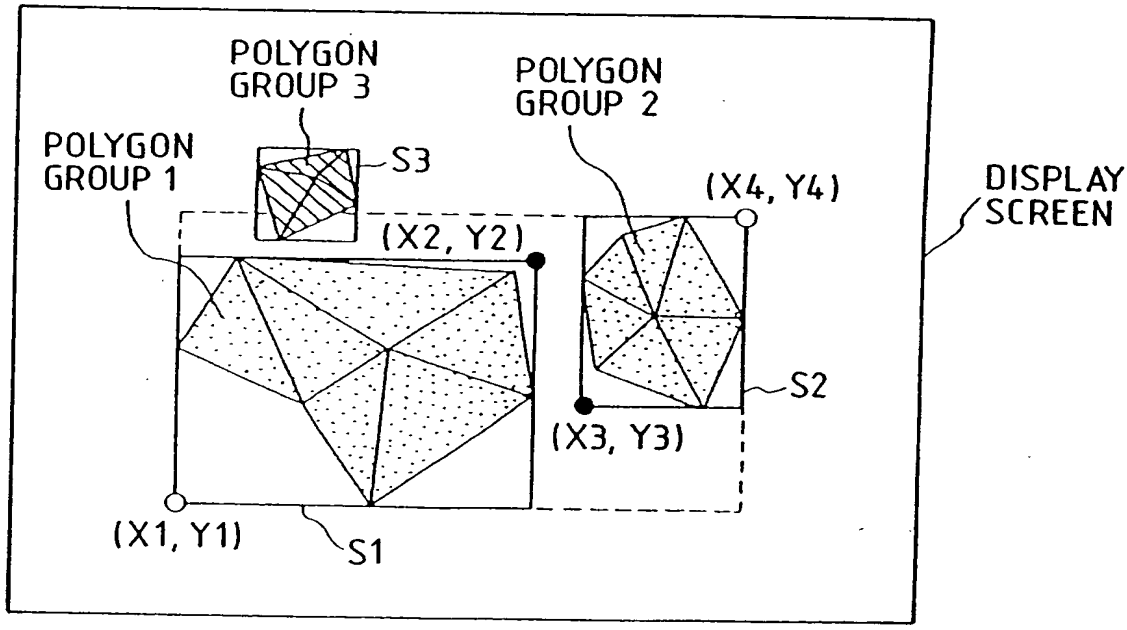

POLYGON
GROUP 3
POLYGON
GROUP 2
POLYGON
GROUP 1
S3
(X4, Y4)
DISPLAY
SCREEN
(X2, Y2)
S2
(X3, Y3)
(X1, Y1)
S1

FIG. 39

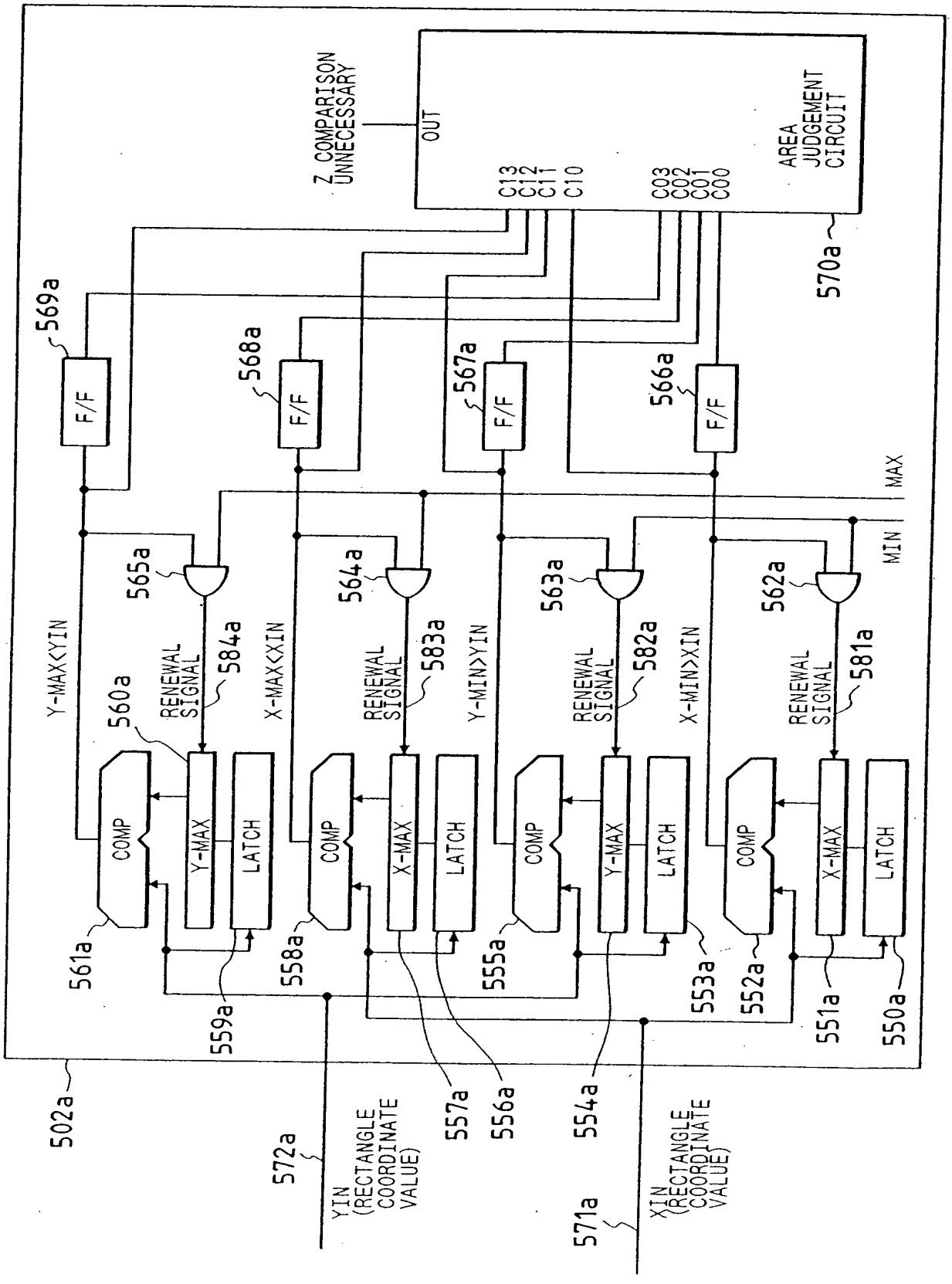

Z COMPARISON UNNECESSARY
OUT
C13
C12
C11
C10
C03
C02
C01
C00
AREA JUDGEMENT CIRCUIT
570a
569a
568a
567a
566a
F/F
MAX
MIN
565a
564a
563a
562a
Y-MAX<YIN
X-MAX<XIN
Y-MIN>YIN
X-MIN>XIN
RENEWAL SIGNAL
584a
583a
582a
581a
560a
COMP
Y-MAX
X-MAX
LATCH
561a
559a
558a
557a
555a
553a
552a
551a
550a
502a
572a
YIN (RECTANGLE COORDINATE VALUE)
556a
554a
571a
XIN (RECTANGLE COORDINATE VALUE)

FIG. 40

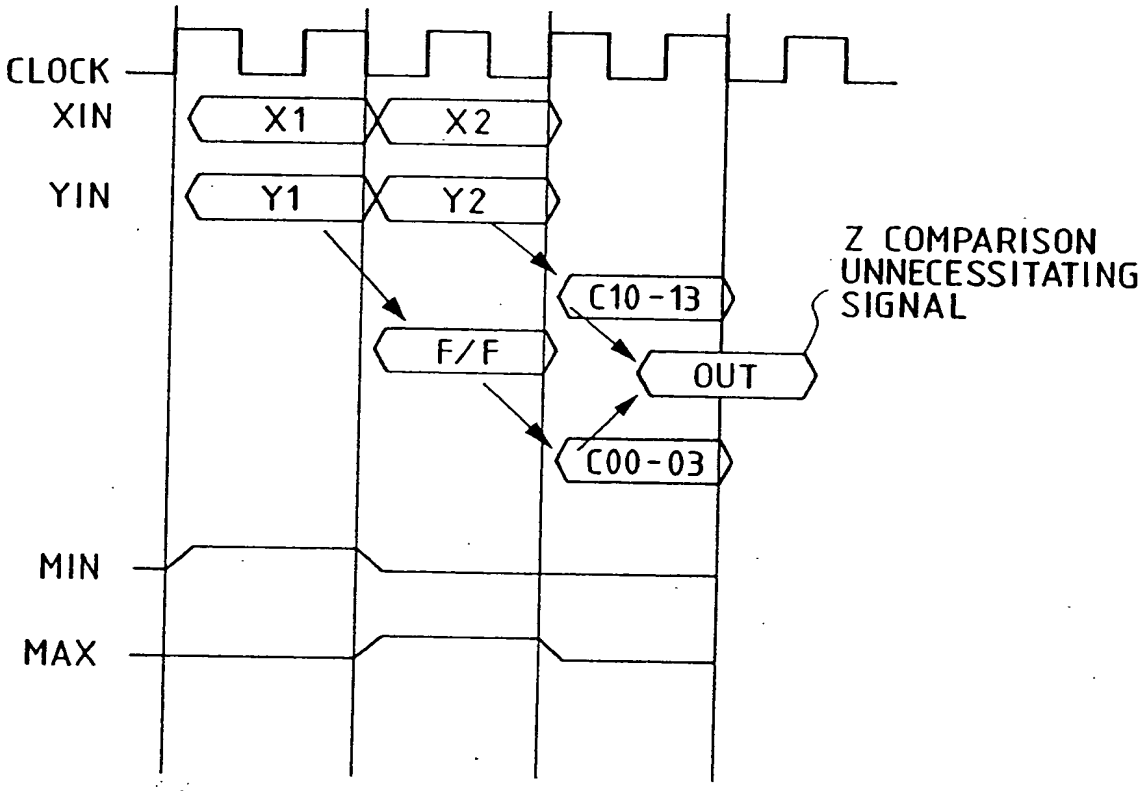

CLOCK
XIN
X1
X2
YIN
Y1
Y2
C10-13
F/F
OUT
C00-03
Z COMPARISON
UNNECESSITATING
SIGNAL
MIN
MAX

FIG. 41

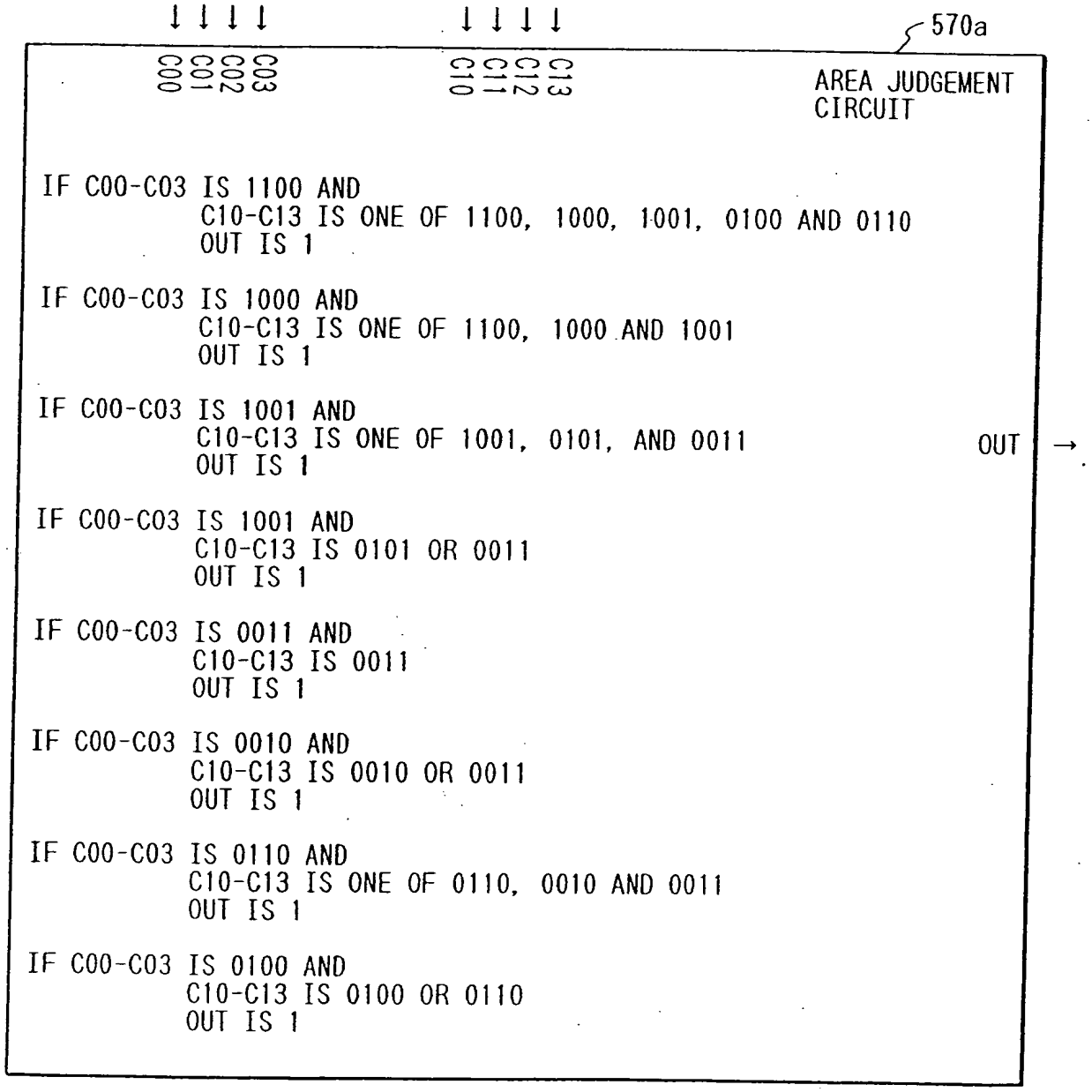

C00
C01
C02
C03
C10
C11
C12
C13
570a
AREA JUDGEMENT CIRCUIT
IF C00-C03 IS 1100 AND
C10-C13 IS ONE OF 1100, 1000, 1001, 0100 AND 0110
OUT IS 1
IF C00-C03 IS 1000 AND
C10-C13 IS ONE OF 1100, 1000 AND 1001
OUT IS 1
IF C00-C03 IS 1001 AND
C10-C13 IS ONE OF 1001, 0101, AND 0011
OUT IS 1
IF C00-C03 IS 1001 AND
C10-C13 IS 0101 OR 0011
OUT IS 1
IF C00-C03 IS 0011 AND
C10-C13 IS 0011
OUT IS 1
IF C00-C03 IS 0010 AND
C10-C13 IS 0010 OR 0011
OUT IS 1
IF C00-C03 IS 0110 AND
C10-C13 IS ONE OF 0110, 0010 AND 0011
OUT IS 1
IF C00-C03 IS 0100 AND
C10-C13 IS 0100 OR 0110
OUT IS 1
OUT

FIG. 42

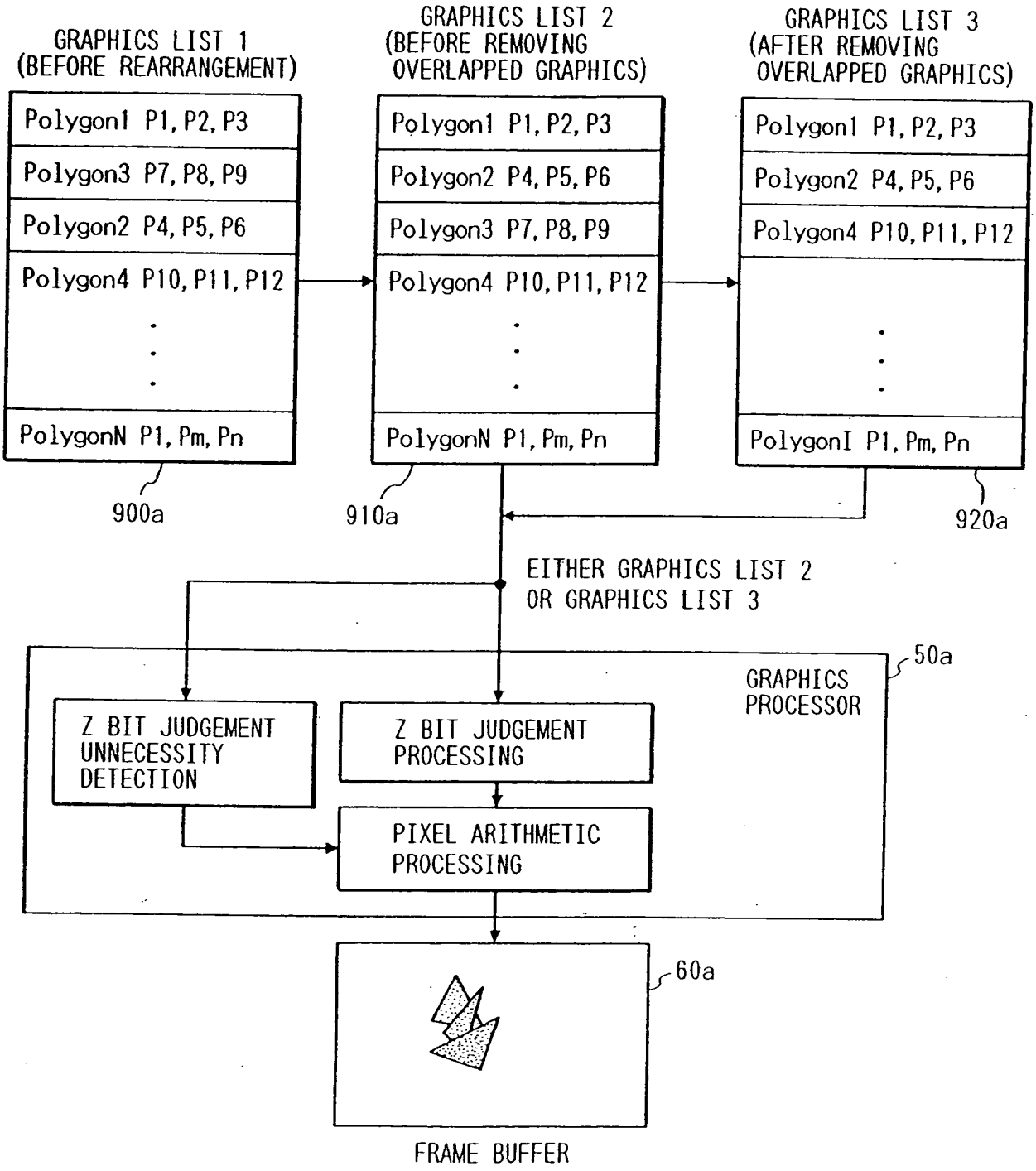

GRAPHICS LIST 1
(BEFORE REARRANGEMENT)
Polygon1 P1, P2, P3
Polygon3 P7, P8, P9
Polygon2 P4, P5, P6
Polygon4 P10, P11, P12
PolygonN P1, Pm, Pn
900a
GRAPHICS LIST 2
(BEFORE REMOVING
OVERLAPPED GRAPHICS)
Polygon1 P1, P2, P3
Polygon2 P4, P5, P6
Polygon3 P7, P8, P9
Polygon4 P10, P11, P12
PolygonN P1, Pm, Pn
910a
GRAPHICS LIST 3
(AFTER REMOVING
OVERLAPPED GRAPHICS)
Polygon1 P1, P2, P3
Polygon2 P4, P5, P6
Polygon4 P10, P11, P12
PolygonI P1, Pm, Pn
920a
EITHER GRAPHICS LIST 2
OR GRAPHICS LIST 3
GRAPHICS
PROCESSOR
50a
Z BIT JUDGEMENT
UNNECESSITY
DETECTION
Z BIT JUDGEMENT
PROCESSING
PIXEL ARITHMETIC
PROCESSING
60a
FRAME BUFFER

FIG. 43

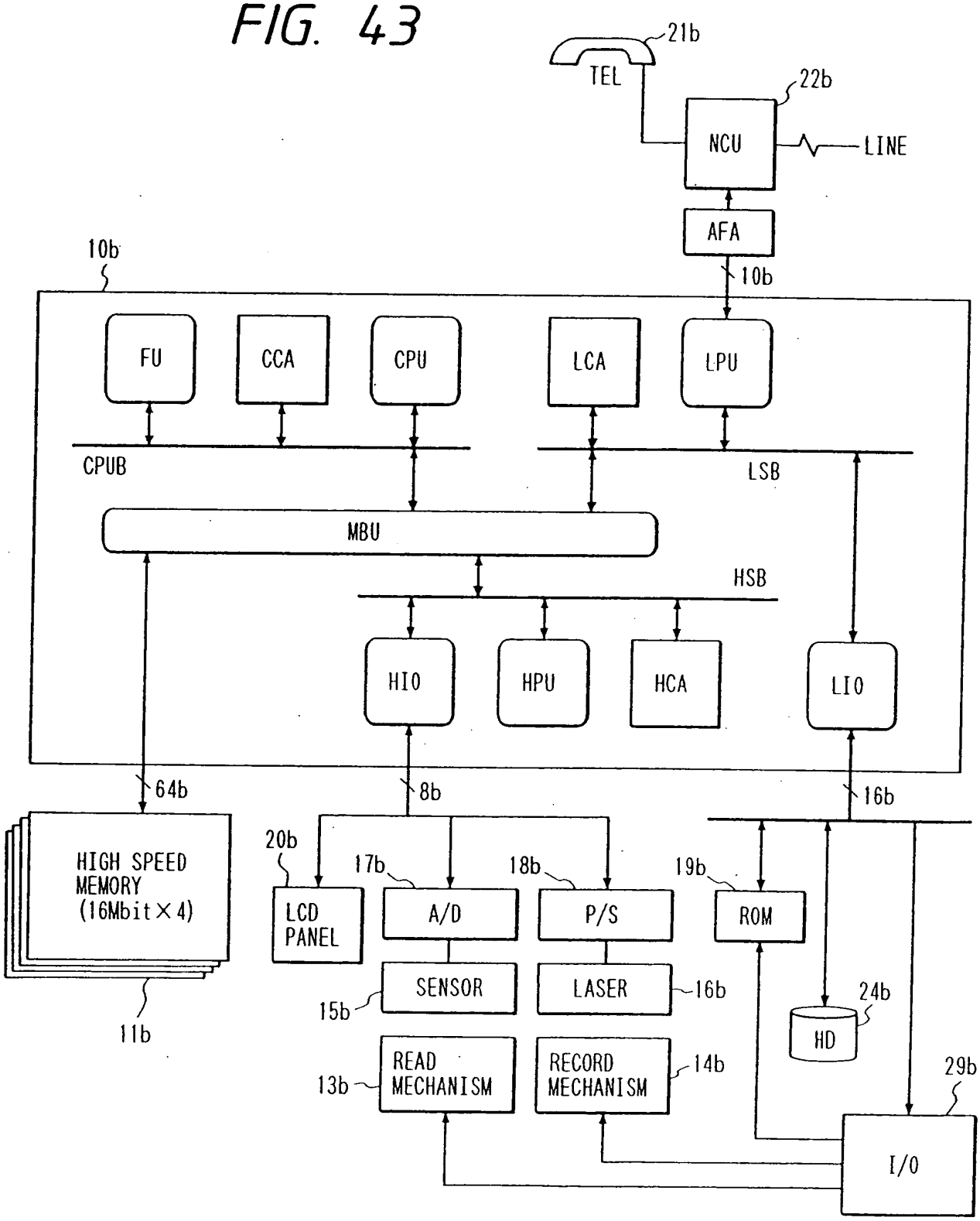

21b
TEL
22b
NCU
LINE
AFA
10b
10b
FU
CCA
CPU
LCA
LPU
CPUB
LSB
MBU
HSB
HIO
HPU
HCA
LIO
64b
8b
16b
HIGH SPEED MEMORY (16Mbit×4)
11b
20b
LCD PANEL
17b
A/D
18b
P/S
19b
ROM
SENSOR
15b
LASER
16b
24b
HD
READ MECHANISM
13b
RECORD MECHANISM
14b
29b
I/O

FIG. 44

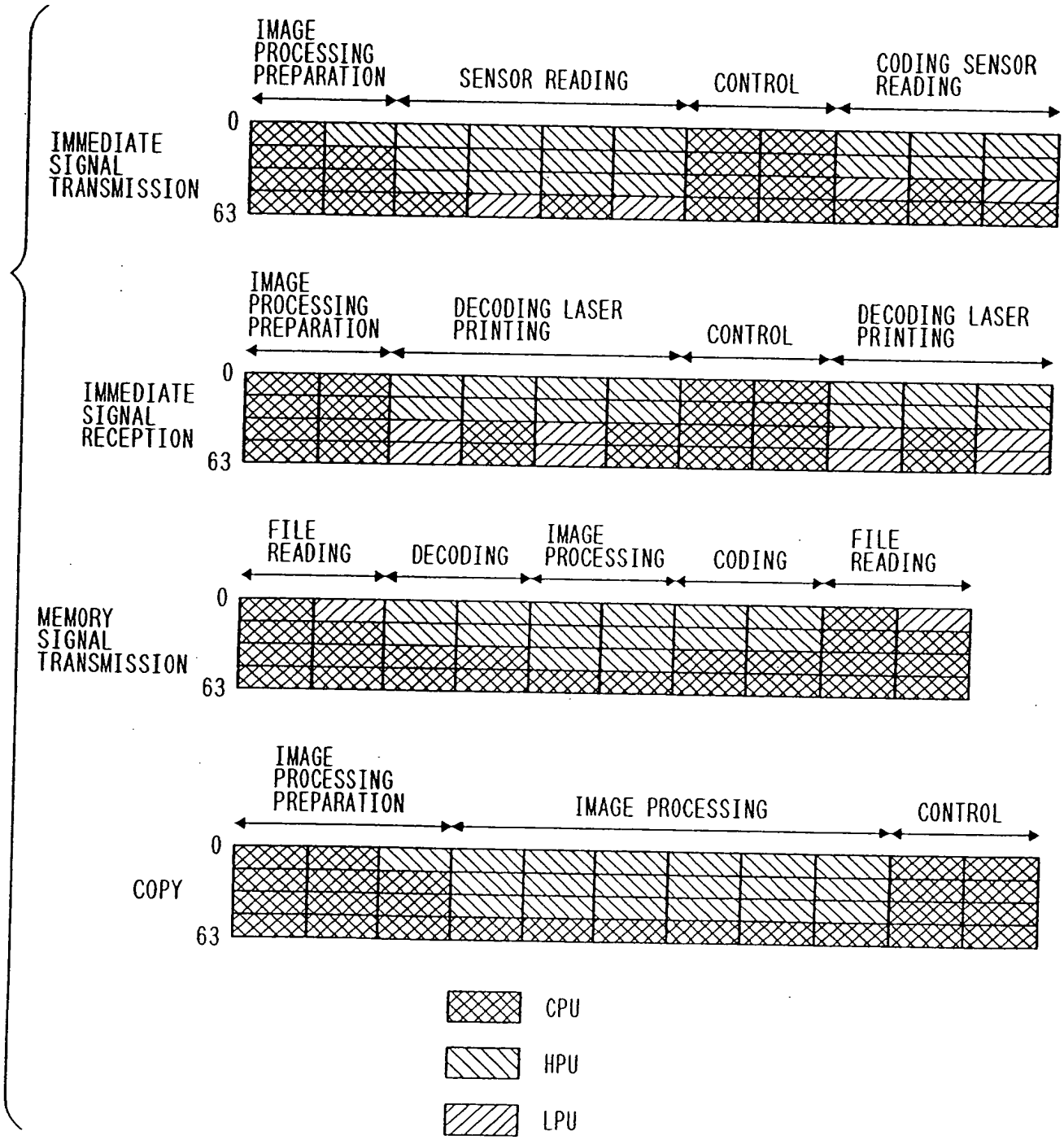


IMAGE PROCESSING PREPARATION
SENSOR READING
CONTROL
CODING SENSOR READING
IMMEDIATE SIGNAL TRANSMISSION
0
63
IMAGE PROCESSING PREPARATION
DECODING LASER PRINTING
CONTROL
DECODING LASER PRINTING
IMMEDIATE SIGNAL RECEPTION
0
63
FILE READING
DECODING
IMAGE PROCESSING
CODING
FILE READING
MEMORY SIGNAL TRANSMISSION
0
63
IMAGE PROCESSING PREPARATION
IMAGE PROCESSING
CONTROL
COPY
0
63
CPU
HPU
LPU

FIG. 45

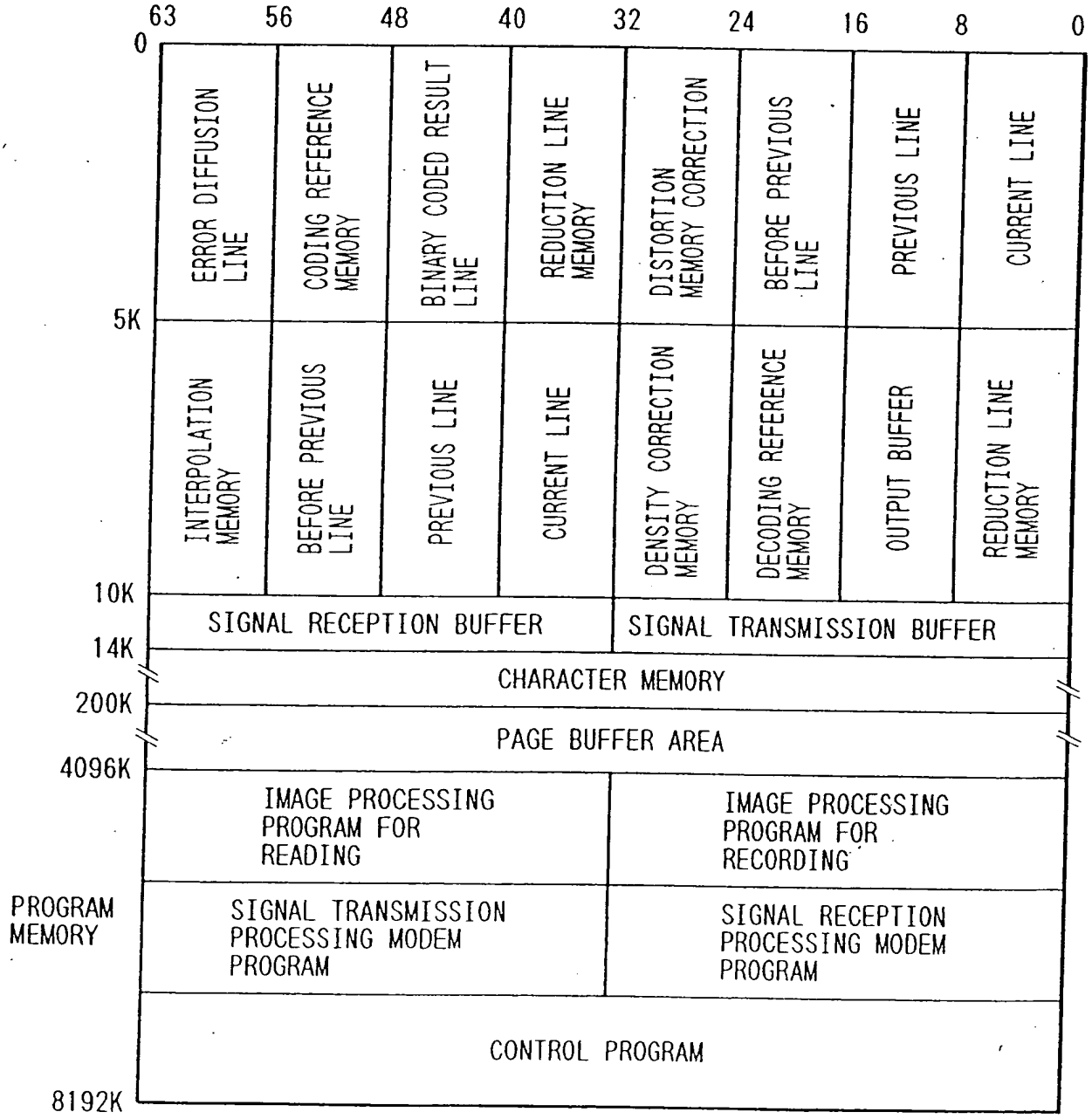

63
56
48
40
32
24
16
8
0
0
ERROR DIFFUSION LINE
CODING REFERENCE MEMORY
BINARY CODED RESULT LINE
REDUCTION LINE MEMORY
DISTORTION MEMORY CORRECTION
BEFORE PREVIOUS LINE
PREVIOUS LINE
CURRENT LINE
5K
INTERPOLATION MEMORY
BEFORE PREVIOUS LINE
PREVIOUS LINE
CURRENT LINE
DENSITY CORRECTION MEMORY
DECODING REFERENCE MEMORY
OUTPUT BUFFER
REDUCTION LINE MEMORY
10K
SIGNAL RECEPTION BUFFER
SIGNAL TRANSMISSION BUFFER
14K
CHARACTER MEMORY
200K
PAGE BUFFER AREA
4096K
PROGRAM MEMORY
IMAGE PROCESSING PROGRAM FOR READING
IMAGE PROCESSING PROGRAM FOR RECORDING
SIGNAL TRANSMISSION PROCESSING MODEM PROGRAM
SIGNAL RECEPTION PROCESSING MODEM PROGRAM
CONTROL PROGRAM
8192K

FIG. 46

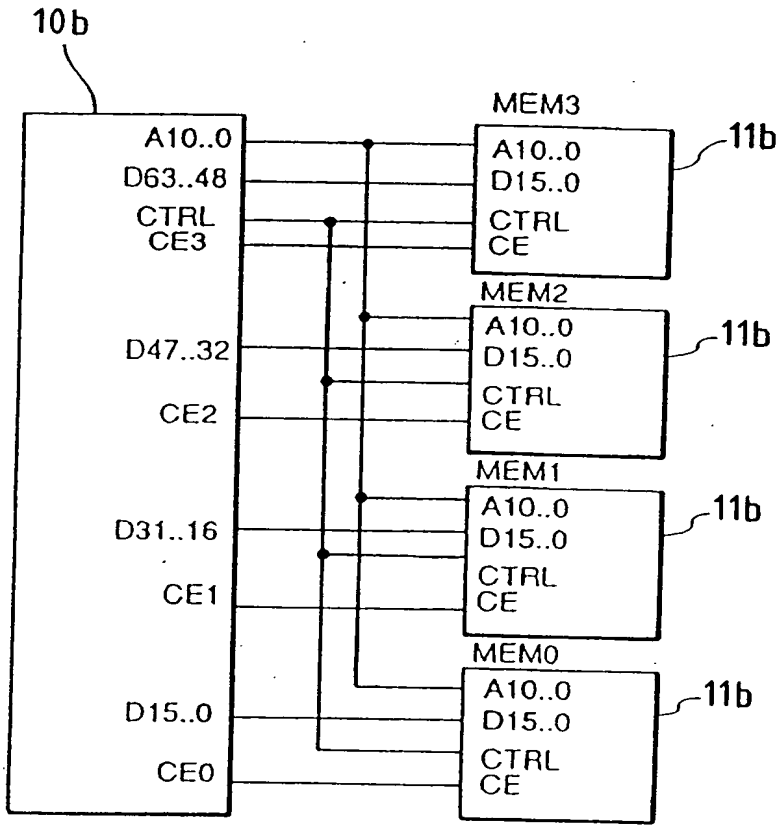
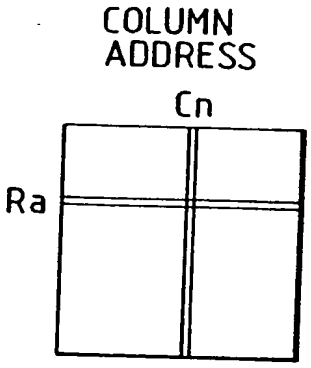
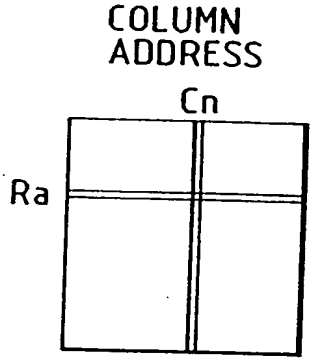

10b
A10..0
D63..48
CTRL
CE3
D47..32
CE2
D31..16
CE1
D15..0
CE0
MEM3
A10..0
D15..0
CTRL
CE
11b
MEM2
A10..0
D15..0
CTRL
CE
11b
MEM1
A10..0
D15..0
CTRL
CE
11b
MEM0
A10..0
D15..0
CTRL
CE
11b

FIG. 47

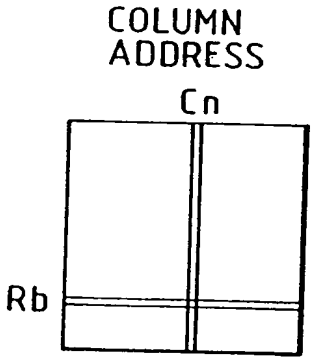
(a) ROW ADDRESS



(b) ROW ADDRESS



(c) ROW ADDRESS



(d) ROW ADDRESS

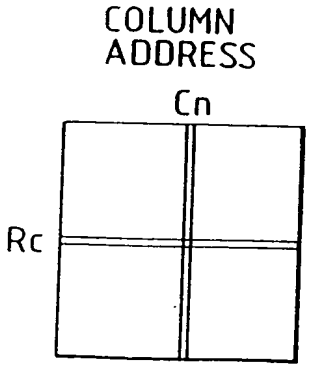
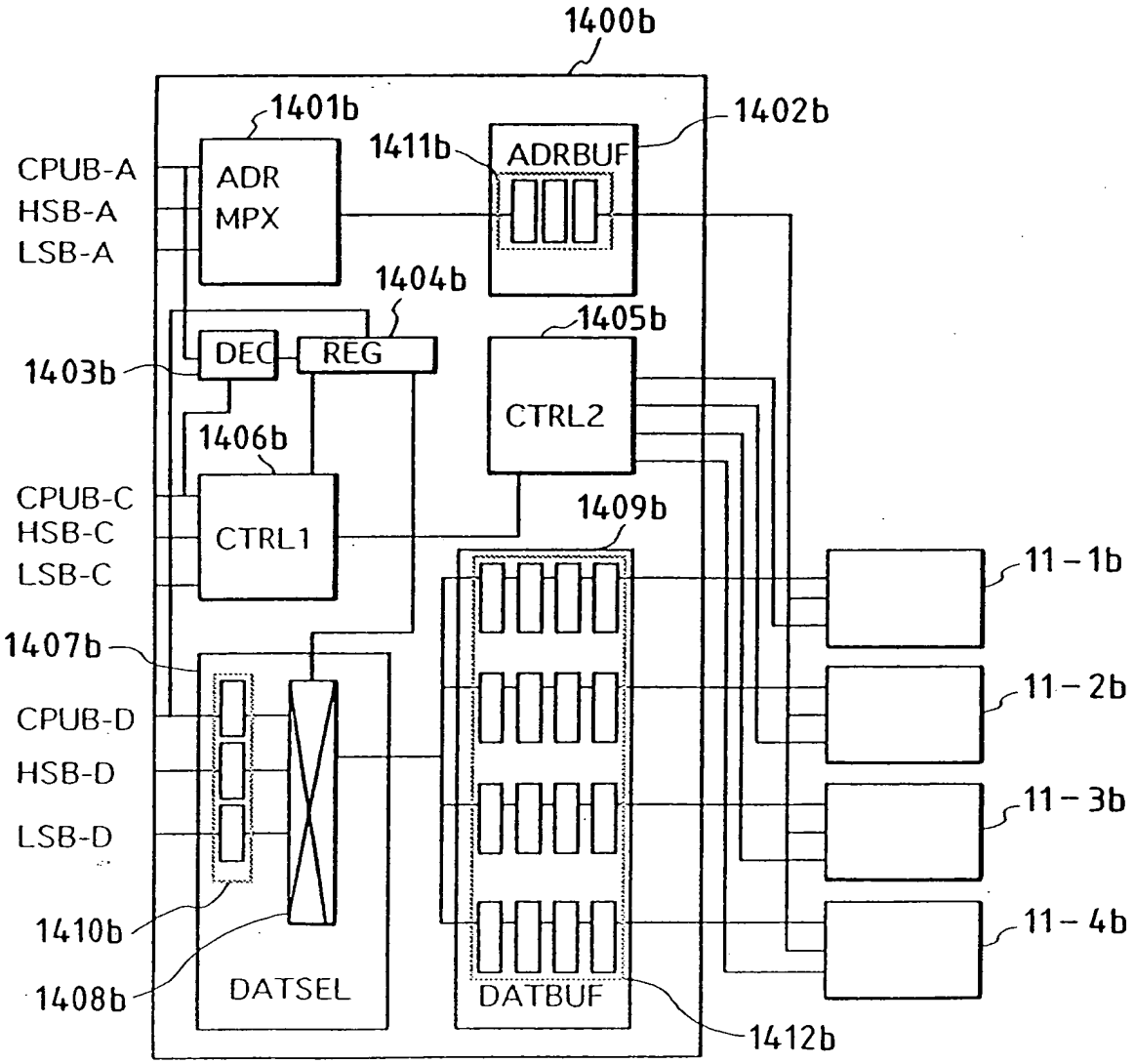


FIG. 48


1400b
1401b
1402b
1411b
ADRBUF
CPUB-A
HSB-A
LSB-A
ADR
MPX
1404b
1405b
DEC
REG
1403b
CTRL2
1406b
CPUB-C
HSB-C
LSB-C
CTRL1
1409b
11-1b
1407b
11-2b
CPUB-D
HSB-D
LSB-D
11-3b
11-4b
1410b
1408b
DATSEL
DATBUF
1412b